HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

CHUYÊN NGÀNH CÔNG NGHỆ PHẦN MỀM

MÔN IOT VÀ ỨNG DỤNG

BÁO CÁO CUỐI KỲ

ĐỀ TÀI: KHÓA CỬA THÔNG MINH NHẬN DIỆN KHUÔN MẶT

Giảng viên hướng dẫn : **Kim Ngọc Bách**
Sinh viên thực hiện : **1. Nguyễn Đức Đạt_B22DCCN195**
2. Phạm Văn Đức_B22DCCN243
3. Trần Gia Hiển_B22DCCN291
4. Nguyễn Xuân Hòa_B22DCCN327
Lớp : **INT14149-20251-05**
Nhóm : **14**

Hà Nội – 2025

MỤC LỤC

DANH MỤC CÁC HÌNH ẢNH

DANH MỤC CÁC TỪ VIẾT TẮT

Từ viết tắt	Tên đầy đủ (Tiếng Anh)	Nghĩa tiếng Việt / Giải thích
ADC	Analog-to-Digital Converter	Bộ chuyển đổi tín hiệu tương tự sang tín hiệu số.
AI	Artificial Intelligence	Trí tuệ nhân tạo.
API	Application Programming Interface	Giao diện lập trình ứng dụng.
CNN	Convolutional Neural Network	Mạng nơ-ron tích chập (Mô hình học sâu dùng để xử lý ảnh).
GPIO	General Purpose Input/Output	Các chân đầu vào/đầu ra đa mục đích trên vi điều khiển.
HTTP	Hypertext Transfer Protocol	Giao thức truyền tải siêu văn bản.
IDE	Integrated Development Environment	Môi trường phát triển tích hợp (Ví dụ: Arduino IDE).
IoT	Internet of Things	Internet vạn vật.
JSON	JavaScript Object Notation	Định dạng dữ liệu dạng văn bản dùng để lưu trữ và trao đổi dữ liệu.
KPIs	Key Performance Indicators	Chỉ số đánh giá hiệu quả hoạt động.
LED	Light Emitting Diode	Diode phát quang (Đèn báo tín hiệu).
MQTT	Message Queuing Telemetry Transport	Giao thức truyền thông điệp theo mô hình Publish/Subscribe.
OTA	Over-The-Air	Phương thức cập nhật phần mềm/firmware từ xa không dây.
PWM	Pulse Width Modulation	Phương pháp điều chế độ rộng xung.
QoS	Quality of Service	Chất lượng dịch vụ (Mức độ tin cậy khi truyền tin trong MQTT).
SPI	Serial Peripheral Interface	Giao diện ngoại vi nối tiếp.
SSL/TLS	Secure Sockets Layer / Transport Layer Security	Giao thức bảo mật lớp truyền tải (Mã hóa dữ liệu).
TCP/IP	Transmission Control Protocol / Internet Protocol	Bộ giao thức điều khiển truyền nhận trên mạng Internet.
UART	Universal Asynchronous Receiver-Transmitter	Bộ truyền nhận dữ liệu nối tiếp bất đồng bộ.

LỜI CẢM ƠN

Đầu tiên, nhóm chúng em xin gửi lời cảm ơn đến Ban Giám hiệu và các thầy cô Học viện Công nghệ Bưu chính Viễn thông đã tổ chức môi trường học tập năng động, tạo điều kiện cho chúng em thực hiện đồ án môn học này.

Đặc biệt, chúng em xin bày tỏ lòng biết ơn sâu sắc đến thầy Kim Ngọc Bách**.** Trong quá trình thực hiện đề tài "Khóa cửa thông minh nhận diện khuôn mặt", thầy đã luôn theo sát, định hướng và tháo gỡ những khó khăn mà chúng em gặp phải, từ việc lựa chọn phần cứng đến việc tối ưu hóa thuật toán nhận diện.

Nhờ sự chỉ bảo của thầy, chúng em đã hiểu sâu hơn về sự kết hợp giữa phần cứng (IoT) và phần mềm (AI/Xử lý ảnh), cũng như tầm quan trọng của các giải pháp nhà thông minh trong đời sống hiện đại. Đây là những kỹ năng và tư duy vô cùng cần thiết cho những kỹ sư tương lai. Chúng em mong muốn Nhà trường sẽ tiếp tục phát triển các môn học mang tính thực tiễn cao như thế này để sinh viên có cơ hội cọ xát và nâng cao tay nghề.

Trong quá trình thực hiện và trình bày báo cáo, do kiến thức và kinh nghiệm còn hạn chế nên khó tránh khỏi những sai sót. Chúng em rất mong nhận được những ý kiến đóng góp quý báu từ thầy để nhóm có thể rút kinh nghiệm và hoàn thiện sản phẩm tốt hơn.

Chúng em xin chân thành cảm ơn!

TÓM TẮT

Hệ thống khóa cửa thông minh nhận diện khuôn mặt.

Đề tài tập trung nghiên cứu, thiết kế và chế tạo hệ thống khóa cửa thông minh sử dụng công nghệ nhận diện khuôn mặt trên nền tảng vi điều khiển ESP 32 CAM. Hệ thống hoạt động dựa trên nguyên lý thu thập dữ liệu hình ảnh từ camera, sau đó áp dụng thuật toán xử lý ảnh để trích xuất đặc trưng và xác thực người dùng. Các chức năng chính bao gồm: tự động nhận diện khuôn mặt để mở khóa, gửi hình ảnh người lạ về máy chủ/điện thoại qua kết nối Wi-Fi, và tích hợp nút nhấn mở cửa thủ công từ bên trong. Qua quá trình thực nghiệm, hệ thống đã chứng minh được khả năng hoạt động ổn định, tốc độ phản hồi nhanh và độ chính xác khá cao trong các điều kiện ánh sáng khác nhau, đáp ứng tốt yêu cầu của một hệ thống an ninh cơ bản cho hộ gia đình.

I. Giới thiệu đề tài

1. Mô tả dự án

- Hiện nay, với sự phát triển của ứng dụng điện toán đám mây và các giao tiếp không dây, việc "thông minh hóa" các hoạt động trong cuộc sống hằng ngày rất được quan tâm và phát triển. Bắt đầu từ nhưng thói quen sử dụng điện thoại thông minh, trợ lý ảo thông minh giúp sắp xếp thời gian biểu hay thông báo lịch hẹn, hay các ưng dụng tài chính thông minh giúp cân đối tài chính cá nhân và gia đình,…. Cho đến những cảnh báo tắc đường, chỉ đường khi tham gia giao thông, tất cả bây giờ đều nằm gọn trong túi quần của bạn. Và tất nhiên, không thể thiếu là hệ thống nhà thông minh, nó đang dần trở thành xu thế của thời đại. Ở đó, với 1 chiếc điện thoại thông minh cũng có thể giúp chúng ta kiểm soát ngôi nhà của mình thông qua các SMS, Email về mọi thứ ta mong muốn như nhiệt độ, bật tắt các thiết bị điện từ xa, kiểm soát tiêu thụ điện năng,… và quan trọng nhất là vấn đề an ninh của ngôi nhà của mình.
- Để giải quyết vấn đề đó, khóa cửa thông minh sinh ra để người dùng có thể bảo vệ tài sản có giá trị và đương nhiên là đáng tin cậy hơn rất nhiều các loại khóa truyền thống. Các khóa thông minh hiện nay sử dụng 3 cơ chế khóa chính là: mở khóa thẻ điện từ, mở khóa nhận diện vân tay, mở khóa bằng mật khẩu. Ngoài các loại khóa thông minh hiện nay, mở khóa bằng "nhận diện khuôn mặt" cũng là đề tài về an ninh bảo mật đang được nghiện cứu và triển khai.
- Không chỉ vậy, khóa thông minh có chức năng chính là tăng cường độ tin cậy về bảo mật, do đó nó được sử dụng vào nhiều hệ thống khác nhau như khóa cửa, khóa phòng, tủ, két sắt,….
- Với hiệu năng làm việc, cùng với độ tin cậy với tính ứng dụng cao, khóa cửa thông minh dần trở thành xu thế tất yếu của cuộc sống hàng ngày của con người.

2. Mục đích, ý nghĩa của dự án

- Nắm được cầu trúc phần cứng, sơ đồ khối, nguyên lý làm việc của mạch điều khiển.
- Tìm hiểu về lập trình với Arduino
- Biết các làm 1 đồ án hoàn chỉnh phục vụ cho việc làm đồ án tốt nghiệp.
- Hệ thống nhận diện được khuôn mặt để mở cửa tự động.
- Có thể giảm sát và điều khiển từ xa quan Internet (Web).
- Có thể cập nhật firmware từ xa (OTA) cho hệ thống để bảo trì, nâng cấp tính năng mà không cần thao tác trực tiếp trên thiết bị.
- Sản phẩm nhỏ gọn mang tính thẩm mỹ cao.
- Tối ưu hóa chi phí, giá thành phù hợp với người tiêu dùng hiện nay.

- Đảm bảo bảo mật, độ chính xác cao, và hoạt động ổn định trong môi trường thực tế.

3. Tổng quan phương hướng

- Trong bối cảnh Công nghệ 4.0 và Internet of Things (IoT) phát triển mạnh mẽ, các thiết bị gia dụng, an ninh và điều khiển thông minh đang trở thành xu hướng tất yếu.
- Hệ thống khóa cửa thông minh "nhận diện khuôn mặt" là một trong ứng dụng tiêu biểu của IoT trong lĩnh vực nhà thông minh (Smart Home) và bảo mật thông minh (Smart Security).
- Cùng với sự phát triển của các công nghệ như:
  + Trí tuệ nhân tạo AI – đặc biệt là nhận diện khuôn mặt.
  + Điện toán đám mây (Cloud Computing).
  + Điện toán biên (Edge Computing).
  + Cập nhật phần mềm từ xa (OTA)
  + Các hệ thống khóa thông minh ngày càng linh hoạt, thông minh và an toàn, giúp con người dễ dàng quản lý, kiểm soát và mở rộng chức năng.
- Phương hướng phát triển:
  + Tích hợp trí tuê nhân tạo giúp hệ thống phân tích dữ liệu từ cảm biến hiệu quả hơn, nhận diện được nhiều khuôn mặt phân loại tốt hơn. Mô hình học sâu tăng cường khả năng dự đoán, nhận diện chính xác hơn.
  + Cải thiện cảm biến thông minh: Cảm biến thông minh trở nên nhỏ gọn và nhạy bén hơn, cho phép tích hợp nhiều loại cảm biển vào 1 thiêt bị duy nhất.
  + Tăng cường bảo mật: Triển khai mã hóa toàn bộ dữ liêu truyền tải, xác thực người dùng đa yếu tố, bổ sung cơ chế phát hiện xâm nhập, cảnh báo tự động qua mail / telegram.
  + Tối ưu khả năng đăng nhập và bảo trì: Hoàn thiện hệ thống cập nhật firmware OTA qua server trung tâm, Cho phép cập nhật đa thiết bị đồng bộ qua mạng.

4. Các thiết bị sử dụng trong hệ thống

- Phần cứng:
  + ESP32 – CAM: Bộ điều khiển tích hợp WIFI và Camera, dùng để thu hình khuôn mặt, xử lý nhận diện và điều khiển khóa cửa.
  + Camera OV2640 (gắn sẵn trên ESP32 – CAM): Chụp hình khuôn mặt truyền cho ESP32 xử lý.
  + Module FTDI (USB to TTL): Dùng nạp trương trình code vào ESP32 – CAM và giao tiếp UART khi code.
  + Relay 5V: Điều khiển solenoid lock hoặc chốt cửa điện tử.
  + Nguồn 5V 2A: Cấp cho ESP32 – CAM và các thiết bị ngoại vi.
  + Khóa điện (12V solenoid): Thiết bị chốt mở cửa.

+ Servo motor: Cơ cấu điều khiển chốt khóa

- Phần mềm:
  + Arduino IDE: Arduino IDE để viết mã cho ESP32. Chương trình viết xong sẽ được nạp vào ESP32 thông qua giao tiếp USB. Arduino IDE có công cụ kiểm tra lỗi và nạp chương trình để giúp đảm bảo rằng chương trình hoạt động ổn định.
  + Arduino OTA Upload: Công cụ cập nhật phần mềm từ xa cho thiết bị qua Wifi.
  + OpenCV + Tensorflow/Keras/Pytorch: Train mô hình nhận diện, trích xuất đặc trưng khuôn mặt.
  + Flask / NodeJs: Xây dựng API nhận ảnh từ ESP32 – CAM, xử lý yêu cầu, quản lý dữ liệu khuôn mặt, trả kết quả.
  + React Js: Thiết kế giao diện quản lý.
  + MongoDB/Firebase: Lưu thông tin người dùng.
  + MQTT Broker: Cổng giao tiếp các thiết IoT với Server.
  + OTA update server: lưu trữ firmware mới và gửi bản cập nhật tới các thiết bị.

II. Cơ sở lý thuyết và các công nghệ áp dụng

1. Lý thuyết về IoT

a. Khái niệm

- IoT (Internet of Things) là mạng lưới các thiết bị vật lý (cảm biến, camera, vi điều khiển, thiết bị gia dụng, v.v.) được kết nối Internet để thu thập, truyền và xử lý dữ liệu.
- Trong dự án: ESP32-CAM là thiết bị IoT, có khả năng thu hình ảnh, gửi dữ liệu nhận diện đến máy chủ và nhận lệnh mở khóa từ xa.

b. Kiến trúc

Tầng	Mô tả	Ứng dụng trong dự án
Perception layer (Tầng cảm nhận)	Gồm các cảm biến, camera, actuator để thu thập dữ liệu môi trường.	ESP32-CAM (camera), relay/servo (mở khóa).
Network layer (Tầng mạng)	Truyền dữ liệu giữa thiết bị và server qua Internet (Wi-Fi, MQTT, HTTP, TCP/IP).	Wi-Fi module trên ESP32, giao tiếp MQTT/HTTP.
Application layer (Tầng ứng dụng)	Phần mềm xử lý, hiển thị, quản lý thiết bị.	Web server / app quản lý nhận diện khuôn mặt.

c. Giao thức truyền dữ liệu

Giao thức	Đặc điểm	Ứng dụng
HTTP/HTTPS	Đơn giản, dễ triển khai, nhưng tốn tài nguyên.	Gửi ảnh lên server AI để nhận diện khuôn mặt.
MQTT (Message Queuing Telemetry Transport)	Nhẹ, nhanh, tối ưu cho IoT, dựa trên publish–subscribe.	Gửi trạng thái "đã nhận diện" / "mở khóa" giữa ESP32 và server.
WebSocket	Kết nối 2 chiều thời gian thực.	Giao tiếp trực tiếp giữa server và web dashboard.

d. Bảo mật

- Xác thực (Authentication): Xác minh danh tính thiết bị/người dùng.
- Mã hóa (Encryption): Bảo vệ dữ liệu trong quá trình truyền (HTTPS, SSL/TLS).
- Phân quyền truy cập (Authorization): Chỉ cho phép người dùng hợp lệ điều khiển khóa.
- Cập nhật OTA an toàn: Firmware mới được kiểm tra chữ ký số trước khi cài đặt.

e. Quản lý thiết bị

- Device provisioning: Gán ID duy nhất cho mỗi ESP32-CAM.
- Device registration: Thiết bị đăng ký với server khi khởi động.
- Remote monitoring: Gửi dữ liệu trạng thái (online/offline, battery, signal...).
- OTA (Over-The-Air update):

f. Xử lý dữ liệu

Kiểu xử lý	Đặc điểm	Ứng dụng
Edge AI (xử lý tại thiết bị)	Xử lý ngay trên vi điều khiển, giảm độ trễ.	ESP32-CAM dùng model nhỏ (nhận diện khuôn mặt đơn giản).
Cloud AI (xử lý trên server)	Gửi dữ liệu (ảnh) lên server để AI xử lý.	Server sử dụng thư viện face_recognition nhận diện khuôn mặt chính xác hơn.

2. Lý thuyết về nhận diện khuôn mặt

a. Khái niệm

- Nhận diện khuôn mặt là quá trình phát hiện, trích xuất đặc trưng, và so sánh khuôn mặt người trong ảnh hoặc video nhằm xác định danh tính hoặc xác thực người dùng.
- Trong dự án IoT, đây là công nghệ cho phép hệ thống xác định ai đang ở trước cửa để ra quyết định mở khóa hay từ chối.

b. Các phương pháp

- ESP32-CAM thực hiện phát hiện khuôn mặt (face detection).
- Nhận diện (recognition) thực hiện trên server bằng thư viện face_recognition.

c. Luồng xử lý tổng quan

1. Ảnh gốc (Matrix WxH) $\xrightarrow{HOG}$ Tọa độ khung (Top, Right, Bottom, Left).
2. Khung ảnh $\xrightarrow{Landmarks + Affine}$ Ảnh mặt thẳng (150x150 px).
3. Ảnh mặt thẳng $\xrightarrow{ResNet-34}$ Vector (List 128 số).
4. Vector A vs Vector B $\xrightarrow{Euclidean}$ Khoảng cách (Số thực 0.0 – 1.0).
5. Khoảng cách $\xrightarrow{Threshold}$ Kết quả (True/False

d. Chi tiết các giai đoạn nhận diện khuôn mặt

- Giai đoạn 1: Tiền xử lý ảnh (Pre-processing)
  Trước khi AI có thể "nhìn", ảnh phải được chuyển đổi thành dạng dữ liệu mà máy tính hiểu được
  + Đầu vào: Một file ảnh (JPEG/PNG) hoặc luồng byte từ ESP32.
  + Xử lý ngầm:
    1. Decoding: Ảnh được giải mã thành một ma trận các điểm ảnh (pixels).
    2. Chuyển đổi hệ màu (Color Space Conversion):
       - OpenCV và nhiều camera đọc ảnh theo chuẩn BGR (Blue-Green-Red).
       - Dlib và face_recognition chỉ làm việc với chuẩn RGB (Red-Green-Blue).
       - Hành động: Hệ thống phải đảo ngược thứ tự các kênh màu của ma trận. Nếu bỏ qua bước này, AI sẽ nhìn màu da người thành màu xanh dương và không nhận diện được.
- Giai đoạn 2: Phát hiện khuôn mặt (Face Detection)
  Mục tiêu: Tìm xem trong bức ảnh khuôn mặt nằm ở đâu (Trả về tọa độ: Top, Right, Bottom, Left).
  + Hàm gọi: face_recognition.face_locations(image, model="hog").
  + Công nghệ lõi: HOG (Histogram of Oriented Gradients) kết hợp với Linear SVM.

  Chi tiết xử lý bên trong:
  1. Grayscale: Ảnh được chuyển tạm thời sang đen trắng.
  2. Tính toán Gradient (Gradient Calculation):
     - Máy tính không nhìn "mũi" hay "mắt". Nó quét từng pixel và so sánh với các pixel xung quanh.

- Nếu pixel này tối, pixel bên cạnh sáng → Có một cạnh (edge).
- Nó vẽ các mũi tên hướng từ tối sang sáng (Gradient vectors).

3. Tạo biểu đồ (Histogram):
   - Nó chia ảnh thành các ô nhỏ (cells, ví dụ 16x16 pixel).
   - Nó tổng hợp hướng của các mũi tên trong ô đó thành một biểu đồ đơn giản hóa.
   - Kết quả: Một khuôn mặt người luôn có cấu trúc HOG đặc trưng (vùng mắt có nhiều gradient ngang, sống mũi có gradient dọc...).
4. Phân loại (Sliding Window & SVM)
   - Một cửa sổ trượt (sliding window) chạy khắp bức ảnh.
   - Tại mỗi vị trí, một thuật toán SVM (Support Vector Machine) sẽ kiểm tra xem biểu đồ HOG ở chỗ này có giống với mẫu khuôn mặt đã học không.
   - Nếu giống → Đánh dấu là khuôn mặt.

- Giai đoạn 3: Căn chỉnh khuôn mặt (Face Alignment & Landmarking)

  Đây là giai đoạn ẩn nằm bên trong hàm face_encodings mà bạn không nhìn thấy trực tiếp. Nó cực kỳ quan trọng để đảm bảo độ chính xác khi người dùng nghiêng đầu.

  + Công nghệ lõi: Ensemble of Regression Trees (để tìm điểm) + Affine Transformation (để xoay ảnh).

  Chi tiết xử lý bên trong:

  1. Tìm 68 điểm mốc (68 Facial Landmarks):
     - Trong cái khung hình chữ nhật tìm được ở Giai đoạn 2, thuật toán sẽ tìm vị trí chính xác của 68 điểm cụ thể: viền cằm (17 điểm), lông mày (10 điểm), mũi (9 điểm), mắt (12 điểm), và môi (20 điểm).
  2. Biến đổi hình học (Affine Transformation):
     - Hệ thống có thể nhận thấy hai mắt của người trong bức ảnh đang bị nghiêng 15 độ, và mặt đang quay sang trái.
     - Nó dùng toán học để xoay (rotate), kéo (scale) và trượt (shear) bức ảnh khuôn mặt đó.
     - Kết quả: Tạo ra một khuôn mặt "chuẩn hóa" (Normalized Face): Mắt nằm ngang, mặt nằm giữa trung tâm, kích thước cố định (thường là 150x150 pixel).

- Giai đoạn 4: Mã hóa đặc trưng (Face Encoding)

  Đây là trái tim của hệ thống, nơi Deep Learning phát huy tác dụng.

  + Hàm gọi: face_recognition.face_encodings(image).

+ Công nghệ lõi: ResNet-34 (Deep Residual Network).

Chi tiết xử lý bên trong:

1. Input: Ảnh khuôn mặt đã được căn chỉnh thẳng thớm từ Giai đoạn 3.
2. Feature Extraction (Trích xuất đặc trưng):
   - Ảnh chạy qua 33 lớp tích chập (Convolutional Layers).
   - Các lớp đầu tiên nhìn thấy đường nét đơn giản (thẳng, cong).
   - Các lớp giữa nhìn thấy các bộ phận (mắt, mũi).
   - Các lớp cuối cùng nhìn thấy các kết cấu vi mô mà mắt thường không thấy được (tỷ lệ khoảng cách xương gò má, độ sâu hốc mắt...).
3. Output (Embedding):
   - Lớp cuối cùng của mạng nơ-ron này là một lớp kết nối đầy đủ (Fully Connected) trả về đúng 128 con số thực.
   - Ví dụ: [-0.124, 0.552, ..., 0.009]
   - ➔ Đây được gọi là Vector đặc trưng (Face Embedding).
   - Cơ chế Triplet Loss: Model này được huấn luyện để sao cho vector của cùng 1 người thì luôn gần nhau, và vector của người khác thì xa nhau

- Giai đoạn 5: So sánh và Nhận diện (Matching)

  Cuối cùng, hệ thống dùng toán học cơ bản để đưa ra kết luận.

  + Hàm gọi: face_recognition.compare_faces.
  + Công nghệ lõi: Khoảng cách Ơ-clit (Euclidean Distance).

  Chi tiết xử lý bên trong:

  1. Hệ thống lấy vector vừa tạo (A) và vector trong CSDL (B).
  2. Nó tính khoảng cách hình học giữa 2 điểm này trong không gian 128 chiều theo công thức:

$$d(A, B) = \sqrt{\sum_{i=1}^{128}(A_i - B_i)^2}$$

  3. Ngưỡng (Thresholding):
     - Mặc định là 0.6.
     - Nếu d < 0.6: Kết luận MATCH (Cùng một người).
     - Nếu d > 0.6: Kết luận MISMATCH (Khác người).

3. Tầm quan trọng của Iot trong bảo mật bằng khóa

- Công nghệ IoT (Internet of Things) đóng vai trò then chốt trong việc phát triển và nâng cao bảo mật của hệ thống khóa thông minh trong thời đại công nghệ số hiện nay. Nhờ khả năng kết nối Internet và trao đổi dữ liệu theo thời gian thực, các thiết bị khóa có thể hoạt động một cách thông minh, linh hoạt và hiệu quả hơn nhiều so với khóa cơ truyền thống. Cụ thể, IoT cho phép người dùng điều khiển, giám sát

và quản lý khóa từ xa thông qua smartphone, máy tính bảng hoặc máy tính, giúp kiểm soát việc ra vào nhà mọi lúc mọi nơi. Khi có hành vi đáng ngờ như truy cập trái phép, cạy khóa hay quên khóa cửa, hệ thống IoT sẽ gửi cảnh báo tức thời đến người dùng, giúp họ chủ động xử lý và bảo vệ tài sản kịp thời.

- Bên cạnh đó, IoT còn cho phép tích hợp khóa thông minh với các thiết bị an ninh khác như camera giám sát, cảm biến chuyển động, hệ thống báo cháy hoặc chuông cửa thông minh, tạo thành một mạng lưới bảo mật thống nhất và toàn diện. Nhờ khả năng thu thập và phân tích dữ liệu, hệ thống có thể học hỏi thói quen người dùng, tự động đóng mở cửa trong những thời điểm phù hợp hoặc nhận diện người dùng hợp lệ qua sinh trắc học (vân tay, khuôn mặt, giọng nói). Điều này không chỉ giúp nâng cao tính an toàn mà còn tăng sự tiện lợi và hiện đại trong quản lý an ninh nhà ở, văn phòng hay cơ sở sản xuất.
- Tóm lại, IoT không chỉ mang đến giải pháp bảo mật tối ưu và thông minh hơn, mà còn mở ra hướng phát triển mới cho ngành công nghiệp an ninh, giúp con người sống an toàn, tiện nghi và chủ động hơn trong kỷ nguyên số.

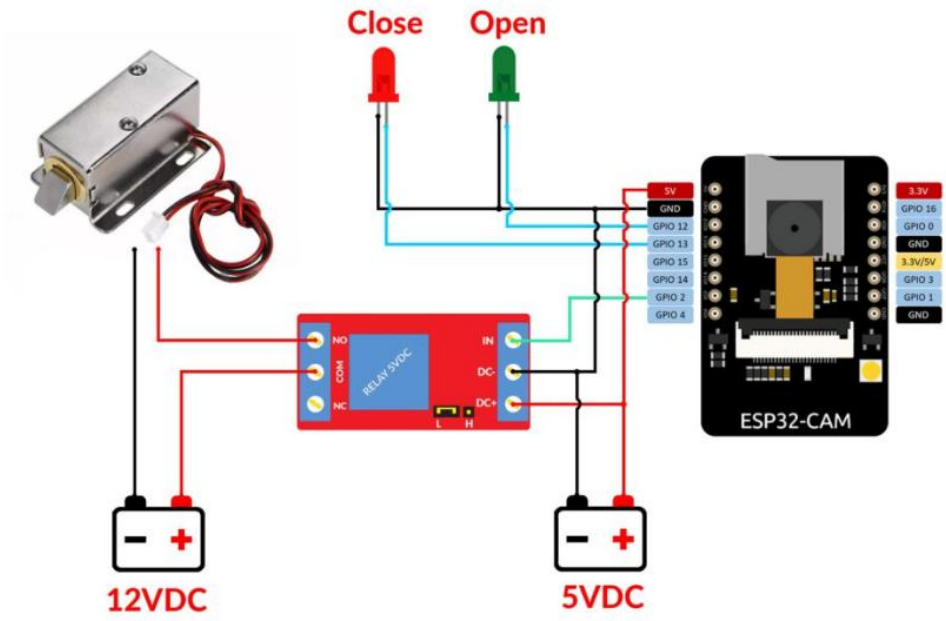
4. Cấu trúc của hệ thống khóa cửa thông minh nhận diện khuôn mặt

Hệ thống khóa cửa thông minh nhận diện khuôn mặt thường bao gồm:

+ Khối cảm biến và nhận dạng là bộ phận đầu vào quan trọng của hệ thống, chịu trách nhiệm thu thập dữ liệu khuôn mặt người dùng. Thành phần này thường sử dụng camera ESP32-CAM để chụp hình khuôn mặt khi người dùng đến gần cửa. Bên cạnh đó, cảm biến chuyển động PIR (Passive Infrared Sensor) được tích hợp để phát hiện chuyển động và kích hoạt camera khi có người xuất hiện, giúp tiết kiệm năng lượng và tăng tính tự động. Để hệ thống hoạt động hiệu quả trong điều kiện ánh sáng yếu hoặc ban đêm, cảm biến hồng ngoại (IR sensor) cũng có thể được sử dụng nhằm hỗ trợ camera nhận diện rõ nét hơn. Dữ liệu hình ảnh thu được từ camera sẽ được truyền đến bộ xử lý trung tâm để tiến hành phân tích, phát hiện và trích xuất đặc trưng khuôn mặt, phục vụ cho quá trình so sánh và nhận diện sau đó.
+ Khối xử lý trung tâm đóng vai trò là "bộ não" của hệ thống, chịu trách nhiệm phân tích, xử lý hình ảnh và ra quyết định mở hoặc khóa cửa. Bộ phận này có thể sử dụng vi điều khiển ESP32, hoặc máy tính nhúng như Raspberry Pi hay Jetson Nano nếu cần hiệu năng cao hơn. Dữ liệu hình ảnh từ camera sẽ được xử lý bằng các thuật toán phát hiện và nhận dạng khuôn mặt như Haar Cascade, MTCNN, FaceNet hoặc mô hình CNN nhẹ, được triển khai thông qua thư viện OpenCV, dlib hoặc TensorFlow Lite. Sau khi trích xuất đặc trưng khuôn mặt, hệ thống sẽ so sánh với dữ liệu khuôn

mặt đã đăng ký sẵn trong cơ sở dữ liệu. Nếu khớp, bộ xử lý trung tâm sẽ gửi tín hiệu điều khiển đến khối chấp hành để mở khóa; ngược lại, nếu không khớp, hệ thống sẽ từ chối truy cập và có thể gửi cảnh báo đến người quản lý. Ngoài ra, khối này cũng có nhiệm vụ ghi lại nhật ký hoạt động và gửi dữ liệu lên máy chủ IoT để lưu trữ và giám sát từ xa.

+ Khối điều khiển và chấp hành là phần thực hiện hành động vật lý của hệ thống, tức là đóng hoặc mở khóa cửa dựa trên kết quả xử lý của khối trung tâm. Thành phần này bao gồm relay hoặc transistor điều khiển để đóng/ngắt dòng điện, khóa điện tử (solenoid lock) hoặc động cơ servo đảm nhiệm chức năng kéo – đẩy chốt cửa. Khi hệ thống xác nhận khuôn mặt hợp lệ, vi điều khiển sẽ gửi tín hiệu kích hoạt relay để mở khóa trong một khoảng thời gian nhất định, sau đó tự động khóa lại nhằm đảm bảo an toàn. Bộ phận này cũng có thể được trang bị cảm biến trạng thái cửa (door sensor) để phát hiện tình trạng cửa đang mở hay đóng, giúp hệ thống giám sát chính xác hơn. Nguồn điện cấp cho khối chấp hành thường đến từ adapter 12V hoặc pin sạc dự phòng, đảm bảo thiết bị hoạt động ổn định và liên tục kể cả khi mất điện.
+ Khối giao tiếp và lưu trữ dữ liệu là phần kết nối hệ thống khóa thông minh với Internet, cho phép người dùng quản lý, theo dõi và điều khiển thiết bị từ xa. Thông thường, hệ thống sẽ sử dụng module WiFi tích hợp sẵn trong ESP32 hoặc các module mở rộng như ESP8266, Bluetooth BLE để kết nối với mạng Internet hoặc điện thoại thông minh. Dữ liệu nhận dạng, nhật ký truy cập, trạng thái khóa và hình ảnh chụp được sẽ được lưu trữ trên máy chủ đám mây (Cloud Server) hoặc cơ sở dữ liệu IoT như Firebase, AWS IoT, hoặc MQTT Broker. Từ đó, người dùng có thể truy cập thông qua ứng dụng di động hoặc giao diện web, để theo dõi hoạt động, thêm hoặc xóa khuôn mặt người dùng, nhận cảnh báo khi phát hiện truy cập bất thường, và thậm chí cập nhật phần mềm hệ thống từ xa. Nhờ có khối giao tiếp này, toàn bộ hệ thống hoạt động theo mô hình IoT hoàn chỉnh – thông minh, an toàn và dễ dàng mở rộng.



Hình 1 Sơ đồ mạch kết nối thiết bị

5. Giao thức truyền tin HTTP

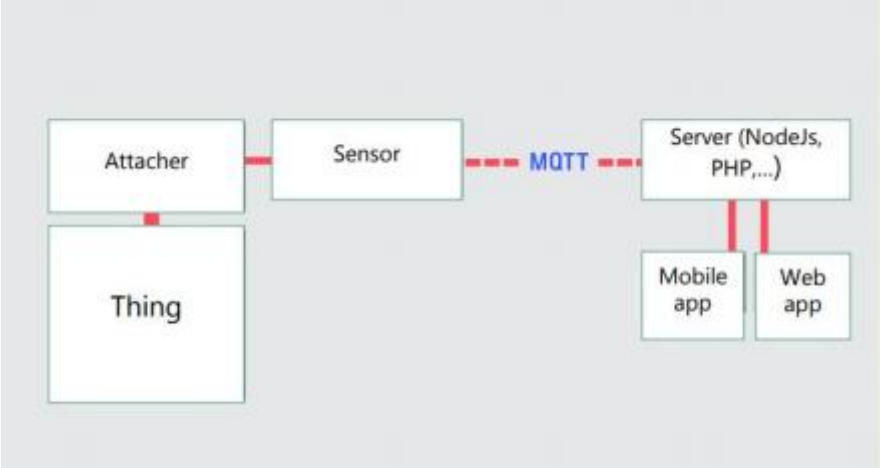
- Khái niệm: Giao thức truyền thông Hypertext Transfer Protocol (HTTP) là một giao thức cơ bản được sử dụng để truy cập và truyền tải dữ liệu trên Internet. Trong lĩnh vực loT, HTTP đóng vai trò quan trọng trong việc kết nối và truyền thông dữ liệu giữa các thiết bị loT và các hệ thống backend.
- HTTP được dùng làm giao thức giao tiếp giữa thiết bị IoT và máy chủ đám mây. Khi camera ESP32-CAM nhận diện được khuôn mặt, dữ liệu hình ảnh hoặc kết quả nhận dạng sẽ được gửi đến máy chủ qua HTTP Request (thường là phương thức POST hoặc PUT). Ngược lại, người dùng có thể điều khiển mở khóa, thêm người dùng mới, hoặc xem nhật ký truy cập thông qua HTTP Response từ máy chủ gửi về. HTTP giúp thiết bị và ứng dụng trao đổi thông tin nhanh chóng, dễ triển khai và tương thích với hầu hết các nền tảng IoT hiện nay.
- Khi người dùng đến gần cửa, camera ESP32-CAM sẽ tự động chụp hình khuôn mặt và gửi dữ liệu hình ảnh đó đến API trên máy chủ thông qua yêu cầu HTTP POST để kiểm tra tính hợp lệ. Máy chủ sau khi nhận dữ liệu sẽ tiến hành so sánh với cơ sở dữ liệu khuôn mặt đã lưu trữ và phản hồi kết quả bằng HTTP Response. Nếu khuôn mặt được xác định là hợp lệ, thiết bị sẽ nhận lệnh mở khóa và kích hoạt relay để điều khiển cơ cấu chốt cửa hoạt động. Ngược lại, nếu khuôn mặt không trùng khớp, hệ thống sẽ giữ nguyên trạng thái khóa và có thể gửi cảnh báo về ứng dụng quản lý thông qua giao thức HTTP

hoặc MQTT, giúp người dùng phát hiện các truy cập bất thường. Bên cạnh đó, người quản trị hệ thống cũng có thể tương tác với máy chủ qua các API RESTful sử dụng HTTP, chẳng hạn như gửi yêu cầu GET để truy vấn danh sách người dùng đã đăng ký, POST để thêm khuôn mặt mới vào cơ sở dữ liệu, hoặc DELETE để xóa quyền truy cập của một người dùng khỏi hệ thống.

- Trong hệ thống khóa cửa thông minh nhận diện khuôn mặt, Client chính là thiết bị IoT như ESP32-CAM hoặc vi điều khiển trung tâm, có nhiệm vụ gửi yêu cầu HTTP đến Server thông qua các API endpoint được định nghĩa sẵn. Khi nhận được yêu cầu, Server (có thể là máy chủ đám mây hoặc máy chủ nội bộ) sẽ xử lý dữ liệu, tiến hành truy vấn cơ sở dữ liệu để kiểm tra thông tin khuôn mặt, đồng thời thực hiện các thao tác xác thực hoặc điều khiển theo yêu cầu của thiết bị. Sau khi hoàn tất xử lý, Server sẽ phản hồi kết quả về cho thiết bị thông qua HTTP Response, trong đó dữ liệu phản hồi được định dạng ở dạng JSON (JavaScript Object Notation) – một cấu trúc dữ liệu nhẹ, dễ phân tích và truyền tải giữa thiết bị IoT và ứng dụng. Nhờ đó, quá trình giao tiếp giữa thiết bị và máy chủ diễn ra nhanh chóng, rõ ràng và thuận tiện cho việc mở rộng hoặc tích hợp thêm các chức năng điều khiển, giám sát từ xa.

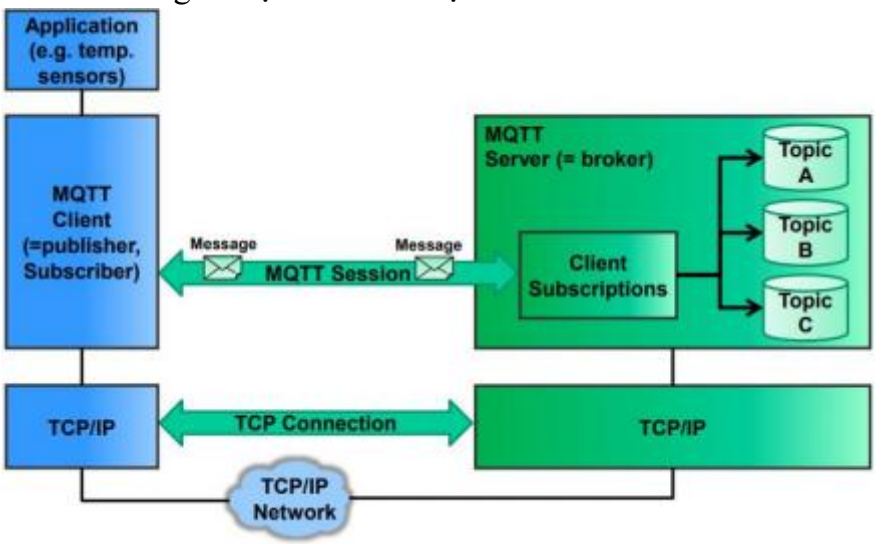
6. Giao thức truyền tin MQTT

- MQTT (Message Queuing Telemetry Transport) là giao thức truyền thông điệp (message) theo mô hình publish/subscribe (cung cấp / thuê bao), được sử dụng cho các thiết bị IoT với băng thông thấp, độ tin cậy cao và khả năng được sử dụng trong mạng lưới không ổn định. Nó dựa trên một Broker (tạm dịch là “Máy chủ môi giới”) “nhẹ” (khá ít xử lý) và được thiết kế có tính mở (tức là không đặc trưng cho ứng dụng cụ thể nào), đơn giản và dễ cài đặt.
- Một số ưu điểm nổi bật của MQTT như: băng thông thấp, độ tin cậy cao và có thể sử dụng ngay cả khi hệ thống mạng không ổn định, tốn rất ít byte cho việc kết nối với server và connection có thể giữ trạng thái open xuyên suốt, có thể kết nối nhiều thiết bị (MQTT client) thông qua một MQTT server (broker). Bởi vì giao thức này sử dụng băng thông thấp trong môi trường có độ trễ cao nên nó là một giao thức lý tưởng cho các ứng dụng IoT.



Hình 2 Mô hình MQTT

- Tính năng và đặc điểm nổi bật



Hình 3 Mô hình MQTT

+ Dạng truyền thông điệp theo mô hình Pub/Sub cung cấp việc truyền tin phân tán một chiều, tách biệt với phần ứng dụng.
+ Việc truyền thông điệp là ngay lập tức, không quan tâm đến nội dung được truyền.
+ Sử dụng TCP/IP là giao thức nền.
+ Tồn tại ba mức độ tin cậy cho việc truyền dữ liệu (QoS: Quality of service)
+ QoS 0: Broker/client sẽ gửi dữ liệu đúng một lần, quá trình gửi được xác nhận bởi chỉ giao thức TCP/IP.
+ QoS 1: Broker/client sẽ gửi dữ liệu với ít nhất một lần xác nhận từ đầu kia, nghĩa là có thể có nhiều hơn 1 lần xác nhận đã nhận được dữ liệu.

    + QoS 2: Broker/client đảm bảo khi gửi dữ liệu thì phía nhận chỉ nhận được đúng một lần, quá trình này phải trải qua 4 bước bắt tay.
    + Phần bao bọc dữ liệu truyền nhỏ và được giảm đến mức tối thiểu để giảm tải cho đường truyền.
- Với những tính năng, đặc điểm nổi bật trên, MQTT mang lại nhiều lợi ích nhất là trong hệ thống SCADA (Supervisory Control And Data Acquisition) khi truy cập dữ liệu IoT.
    + Truyền thông tin hiệu quả hơn.
    + Tăng khả năng mở rộng.
    + Giảm đáng kể tiêu thụ băng thông mạng.
    + Rất phù hợp cho điều khiển và do thám.
    + Tối đa hóa băng thông có sẵn.
    + Chi phí thấp.
    + Rất an toàn, bảo mật.
    + Được sử dụng trong các ngành công nghiệp dầu khí, các công ty lớn như Amazon, Facebook, ...
    + Tiết kiệm thời gian phát triển.
    + Giao thức publish/subscribe thu thập nhiều dữ liệu hơn và tốn ít băng thông hơn so với giao thức cũ

7. Websocket:

- Khái niệm: Websocket là 1 giao thức truyền thông cung cấp các kênh liên lạc song song hoàn toàn qua kết nối TCP duy nhất giữa máy chủ với máy khách. Không giống như HTTP truyền thông theo mô hình request – response, giao thức này cho phép giao tiếp 2 chiều. Điều này có nghĩa là máy khách và máy chủ có thể gửi dữ liệu cho nhau bất cứ lúc nào, giúp dữ liệu gửi đi nhanh chóng mà không cần tải lại trang web.
- Cách hoạt động:
    + Client yêu cầu mở kết nối Websocket tới server.
    + Server chấp nhận kết nối và thiết lập 1 kênh giao tiếp 2 chiều.
    + Sau khi kết nối được thiết lập, cả 2 bên có thể gửi nhận dữ liệu qua nhau mà không cần các kết nối HTTP nào.
    + Sau khi hoàn thành trao đổi dữ liệu, kết nối vẫn sẽ được gửi cho đến khi 1 trong 2 bên ngắt kết nối.

8. Firebase

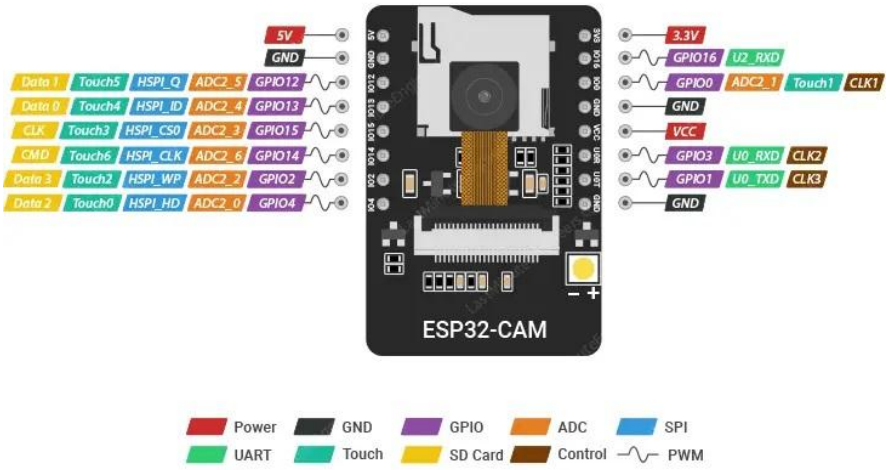
- Khái niệm: Firebase trong IoT là một nền tảng được sử dụng để hỗ trợ lưu trữ và quản lý dữ liệu của các thiết bị IoT trên đám mây. Firebase cung cấp một loạt các dịch vụ, chẳng hạn như cơ sở dữ liệu thời gian thực, lưu trữ tệp, phân tích, và khả năng xác thực người dùng, giúp các hệ thống IoT có thể lưu trữ, xử lý và đồng bộ dữ liệu dễ dàng giữa các thiết bị và ứng dụng.

- Các khía cạnh chính về việc sử dụng Firbase trong dự án IoT:
  + Firebase Realtime Database: là một giao thức được xây dựng dựa trên WebSocket và HTTP để cung cấp kết nối dữ liệu liên tục theo thời gian thực, là một cơ sở dữ liệu NoSQL, nơi mà dữ liệu được lưu trữ dưới dạng JSON và đồng bộ hóa theo thời gian thực. Điều này rất phù hợp với các ứng dụng IoT, vì các thiết bị IoT thường cần gửi và nhận dữ liệu trong thời gian thực.
  + Firebase Cloud Messaging (FCM): cho phép các thiết bị IoT gửi và nhận thông báo thông qua dịch vụ đám mây của Firebase. Ví dụ, một thiết bị IoT có thể phát hiện sự cố (như phát hiện khói hoặc khí gas vượt ngưỡng) và gửi thông báo tới người dùng ngay lập tức.
  + Firebase Hosting: có thể được sử dụng để lưu trữ ứng dụng web, trang điều khiển các thiết bị IoT, giúp quản lý và điều khiển thiết bị từ xa qua giao diện người dùng (UI).

9. Các thiết bị sử dụng trong hệ thống

a. ESP32 – CAM

- ESP32-CAM có một camera kích thước nhỏ, rất cạnh tranh trong ngành, giống như mô-đun chính, mô-đun này có thể được xử lý công việc độc lập, module có kích thước nhỏ gon chỉ 40 x 27 x12 mm, dòng nghỉ chỉ 6mA.
- ESP-32CAM có thể được sử dụng rộng rãi trong các ứng dụng IoT khác nhau, thích hợp cho thiết bị thông minh gia đình, điều khiển không dây công nghiệp, giám sát không dây kiểm soát, nhận dạng không dây QR, tín hiệu hệ thống định vị không dây…Nó là một giải pháp lý tưởng cho các ứng dụng IoT
- Mạch thu phát Wifi BLE ESP32 này là mạch chính hãng AI – Thinker có chất lượng độ ổn định và độ bền rất cao, sử dụng camera OV2640 chất lượng cao hình ảnh sắc nét, không nhiễu sọc, không xảy ra tình trạng treo khi hoạt động do sử dụng ic cấp nguồn chất lượng cao.
- Mạch thu phát Wifi BLE ESP32-CAM Ai-Thinker này có thể sử dụng Arduino IDE để biên dịch và viết code, được hỗ trợ mạnh mẽ từ cộng đồng.



Hình 4 Sơ đồ chân mạch ESP 32 CAM

- Cấu tạo:
  + Power (Chân nguồn): Có hai chân nguồn là 5V và 3V3. Chúng ta có thể cấp nguồn cho ESP32-CAM qua một trong hai chân này. Vì nhiều người dùng đã gặp phải sự cố khi cấp nguồn cho thiết bị ở mức 3,3V, nên ESP32-CAM luôn được cấp nguồn qua chân 5V. Chân VCC thường xuất ra 3,3V từ bộ điều chỉnh điện áp trên bo mạch. Tuy nhiên, nó có thể được cấu hình để xuất ra 5V bằng cách sử dụng liên kết Zero-ohm gần chân VCC.
  + GND: Nối đất
  + GPIO: Trên ESP32-S có tổng 32 chân GPIO, nhưng có một số chân được dùng nội bộ cho PSRAM và máy ảnh nên chúng ta chỉ còn lại 10 chân có thể sử dụng. Trong đó, mỗi chân lại có 1 nhiệm vụ ngoại vi khác nhau, chẳng hạn như SPI, UART, ADC hoặc Touch.
  + UART: Trên thực tế, ESP32-S có hai giao diện UART là UART0 và UART2. Tuy nhiên, chân RX (GPIO 16) của UART2 bị hỏng, khiến chúng ta chỉ có thể dùng UART0 trên ESP32-CAM (GPIO 1 và GPIO 3). Ngoài ra, do ESP32-CAM thiếu cổng USB nên các chân này phải dùng để bật đèn flash cũng như kết nối với các thiết bị UART như GPS, cảm biến vân tay, cảm biến khoảng cách,… tùy theo nhu cầu người dùng.
  + MicroSD: Dùng để kết nối thẻ nhớ microSD. Nếu không sử dụng thẻ nhớ microSD, bạn có thể sử dụng các chân này làm đầu vào và đầu ra thông thường.
  + ADC: Các chân ADC2 được trình điều khiển WiFi sử dụng nội bộ nên chúng không thể được sử dụng khi đang bật Wi-Fi.

+ Touch: ESP32-CAM có 7 GPIO cảm ứng điện dung. Khi tải điện dung (chẳng hạn như ngón tay người) ở gần GPIO, ESP32 sẽ phát hiện sự thay đổi điện dung.
+ SPI: ESP32-CAM chỉ có một SPI (VSPI) ở chế độ phụ và chế độ chính.
+ PWM: ESP32-CAM có 10 kênh (tất cả các chân GPIO) PWM được điều khiển bởi bộ điều khiển PLC. Đầu ra PWM có thể được sử dụng để điều khiển động cơ kỹ thuật số và đèn LED.

- Nguyên lý hoạt động:
  + Các vai trò chính của ESP 32 CAM:

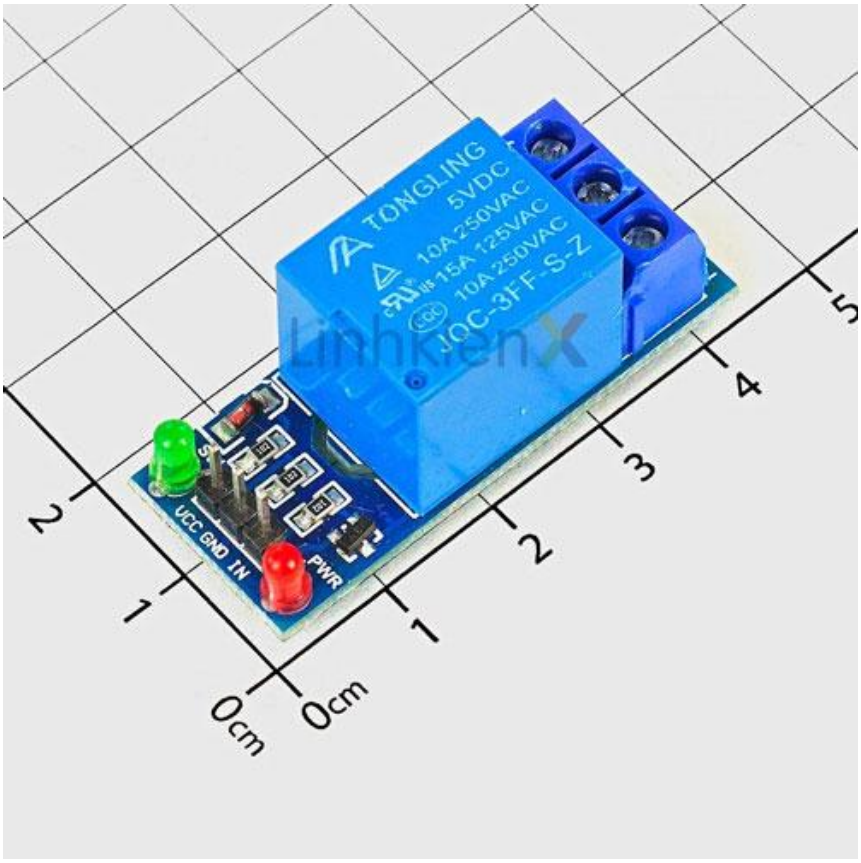
Chức năng	Mô tả
Thu thập hình ảnh	Sử dụng module camera (OV2640) để chụp khuôn mặt của người đứng trước cửa.
Kết nối Wifi	Duy trì kết nối giao tiếp với mạng Wifi cục bộ để giao tiếp với máy chủ server cục bộ.
Giao tiếp IoT	Gửi dữ liệu ảnh lên server để nhận diện, và nhận lệnh điều khiển từ server (thường qua giao thức MQTT / HTTP).
Điều khiển thiết bị	Kích hoạt Relay Module khi nhận được lệnh mở khóa hợp lệ từ server, từ đó cấp điện cho khóa.
Quản lý nguồn & trạng thái	Quản lý trạng thái nguồn điện, có thể chuyển sang chế độ ngủ, để tiết kiệm điện khi không có hoạt động.

  + Nguyên lý hoạt động:
    - Hệ thống khóa cửa thông minh sử dụng ESP 32 CAM thường sử dụng các mô hình xử lý ở phía server do khả năng tính toán hạn chế của ESP 32, giúp đảm bảo tốc độ và độ chính xác cao hơn.
    - Quá trình hoạt động:
      - B1: Kích hoạt:
        - ESP 32 CAM được cấp nguồn và kết nối Wifi thành công.
        - Camera được kích hoạt thông qua nút bấm và bắt đầu thu nhận hình ảnh.
        - Khi có kích hoạt, ESP 32 CAM chụp 1 bức ảnh có độ phân giải phù hợp chứa khuôn mặt.
      - B2: Truyền dữ liệu đến Server:
        - ESP 32 CAM nén bức ảnh đã chụp.

        - Ảnh được gửi lên server thông giao thức HTTP POST hoặc giao thức MQTT (dùng base64 để nhúng ảnh vào payload).
    - B3: Xử lý nhận diện:
        - Nhận diện:
            - ✓ Server nhận ảnh, dùng các model để thực hiện:
            - ✓ Phát hiện khuôn mặt: dùng hàm face_locations() để phát hiện khuôn mặt trong ảnh.
            - ✓ Trích xuất đặc trưng: chuyển ảnh khuôn mặt thành các vector.
            - ✓ So sánh & nhận dạng: so sánh các đặc trưng này với các đặc trưng của ảnh trong database.
        - Quyết định:
            - ✓ Nếu khớp, server xác định quyền mở cửa.
            - ✓ Nếu không khớp, đưa ra thông báo đề cả thiết bị lẫn người dùng.
    - B4: Điều khiển cơ cấu chấp hành:
        - Gửi lệnh: Nếu truy cập hợp lệ, server ngay lập tức gửi lệnh mở cửa đến cho ESP 32 CAM (thông qua MQTT).
        - Kích hoạt Relay: ESP 32 CAM nhận lệnh, xử lý và gửi tín hiệu logic tới relay module.
        - Mở khóa: Relay kích hoạt, đóng mạch, cấp nguồn cho khóa trong khoảng thời gian nhất định.
    - B5: Phản hồi và cảnh báo:
        - Cập nhật trạng thái: ESP 32 CAM gửi xác nhận lên server.
        - Thông báo người dùng: Server đồng thời gửi thông báo real time đến cho người dùng về xự kiện xảy ra.

b. Relay Module (1 kênh)

- Module Relay là một thiết bị điện tử giúp chuyển mạch bằng cách sử dụng tín hiệu điều khiển điện áp thấp để điều khiển các thiết bị điện công suất cao. Relay đóng vai trò như một công tắc điều khiển từ xa, cho phép bật/tắt các thiết bị điện mà không cần tiếp xúc trực tiếp. Nhờ đó, relay được sử dụng rộng rãi trong các hệ thống tự động hóa, nhà thông minh và IoT.
- Module Relay 1 kênh: Điều khiển một thiết bị duy nhất, thường được sử dụng trong các ứng dụng đơn giản như bật/tắt đèn, quạt hoặc bơm nước.



Hình 5 Relay 1 kênh 12V

- Relay hoạt động dựa trên nguyên tắc của điện từ trường. Khi dòng điện chạy qua cuộn dây, nó tạo ra một từ trường hút hoặc nhả tiếp điểm bên trong relay. Nhờ đó, relay có thể đóng hoặc mở mạch điện đầu ra, giúp điều khiển các thiết bị điện lớn bằng tín hiệu điều khiển điện áp thấp.
- Cấu tạo:
  + Cuộn dây (Coil): Là bộ phận tạo từ trường khi có dòng điện chạy qua, quyết định khả năng đóng/mở của relay.
  + Tiếp điểm (Contacts): Thành phần quan trọng giúp relay đóng/mở mạch điện, điều khiển trực tiếp thiết bị.
  + Diode bảo vệ: Giúp ngăn chặn dòng ngược từ cuộn dây, tránh làm hỏng vi điều khiển.
  + Mạch cách ly (Optocoupler, Transistor): Đảm bảo sự an toàn và ổn định của tín hiệu điều khiển, giảm thiểu nhiễu điện.
  + LED báo trạng thái: Hiển thị trạng thái hoạt động của relay, giúp người dùng dễ dàng kiểm tra.
- Nguyên lý hoạt động:
  + Trạng thái tắt (OFF): Khi ESP 32 CAM gửi tín hiệu LOW (0V), cuộn dây của Relay không có điện. Tiếp điểm ở trạng

thái mặc định: COM nối với NC, hở NO. Khóa điện không được cấp nguồn.

+ Trạng thái mở (ON): Khi ESP 32 CAM gửi tín hiện HIGH (5V), dòng điện chạy qua cuộn dây tạo ra từ trường. Từ trường này hút các tiếp điểm cơ học làm thay đổi trạng thái: COM ngắt kết nối với NC chuyển sang kết nối với NO. Khóa điện được cấp nguồi 12V và mở cửa.

c. Mạch chuyển nguồn 12V sang 5V

Hình 6 Mạch chuyển nguồn 12V -> 5V, 3V

- Thông số kỹ thuật:
  + Điện áp đầu vào: 6 ~ 12V (điện áp đầu ra phải lớn hơn điện áp đầu vào 1V).
  + Điện áp đầu ra: 3V, 5V, 12V (đầu vào 12V được chuyển đổi trực tiếp thành đầu ra) (sai số +- 0.2).
  + Đầu vào và đầu ra sử dụng nhiều đầu cắm để dễ sử dụng và kết nối.
  + Kích thước: 59.5 * 26.5 mm.
  + Có đèn led báo nguồn vào (đỏ).
  + Jack DC 5.5 * 2.1 mm.

d. Khóa điện tử

- Khóa chốt điện từ LY-03, có chức năng hoạt động như một ổ khóa cửa sử dụng Solenoid để kích đóng mở bằng điện, được sử dụng nhiều trong nhà thông minh hoặc các loại tủ, cửa phòng, cửa kho,… khóa sử dụng điện áp 12VDC, là loại thường đóng (cửa đóng) với chất lượng tốt, độ bền cao. Khóa chốt điện từ này có thể sử dụng chung với các mạch chức năng tạo thành một hệ thống thông minh.

Hình 7 Khóa điện tử 12V – 0.6 A

- Thông số kỹ thuật
  + Vật liệu: Thép không gỉ
  + Nguồn điện: 12V DC
  + Dòng điện làm việc: 0.6A
  + Công suất: 9.6W
  + Yêu cầu nguồn cấp: 12VDC/1A
  + Kích thước: L54 x D38 x H28
  + Thời gian cấp nguồn: Nhỏ hơn 10s
- Nguyên lý hoạt động
  + Trạng thái khóa: Khóa không được cấp điện, lò so bên trong đẩy chốt kim loại ra ngoài, giữ khóa ở trạng thái đóng.
  + Trạng thái mở: Khi nhận lệnh mở cửa của ESP 32 CAM, Relay đóng mạch, cấp nguồn 12V cho cuộn dây Solenoid bên trong khóa. Dòng điện chay trong cuôn dây tạo ra từ trường, từ trường này hút chốt kim loại vào trong, giữ trạng thái mở trong 1 khoảng thời gian nhất định.

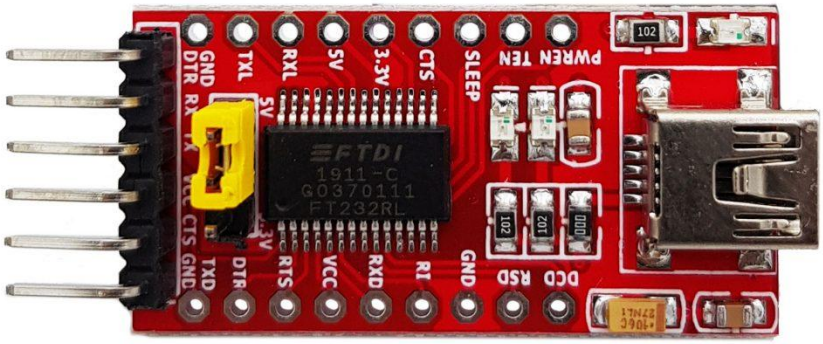
e. Nút bấm 4 chân

Hình 8 Nút bấm 4 chân

- Thông số kỹ thuật:
  + Dòng điện định mức: 12V DC – 0.1A
  + Điện trở tiếp điểm: <= 0.03 ôm.
  + Quy cách: Nhấn nhả 4 chân.
  + Chiều cao: 14mm.
  + Kích thước: 6 * 6 * 14 mm
- Tác dụng:
  + Kích hoạt cấp nguồn cho ESP 32 CAM

f. FTDI232 USB to TTL conveter

- Mạch chuyển USB UART TTL FT232RL sử dụng IC FT232RL từ chính hãng FTDI, mạch được thiết kế nhỏ gọn nhưng vẫn ra chân đầy đủ, rất dễ sử dụng với mọi hệ điều hành Windows, Mac, Linux.
  + Chip có sẵn ổn áp và dao động tích hợp bên trong, hoạt động rất ổn định so với các dòng chip USB to serial khác
  + Mạch có thể hoạt động ở 2 chế độ 5v hoặc 3v3, bằng cách thiết lập trên jumper trên mạch
  + Chân cắm ra gồm 2 loại theo chuẩn FTDI (phù hợp với Arduino) và chuẩn UART thường, được ký hiệu rõ ràng trên mạch. Đầu vào sử dụng loại USB B mini.
  + Ngoài ra, trên mạch có sẵn 2 led cho tín hiệu TX và RX, giúp theo dõi trực tiếp trạng thái tín hiệu.



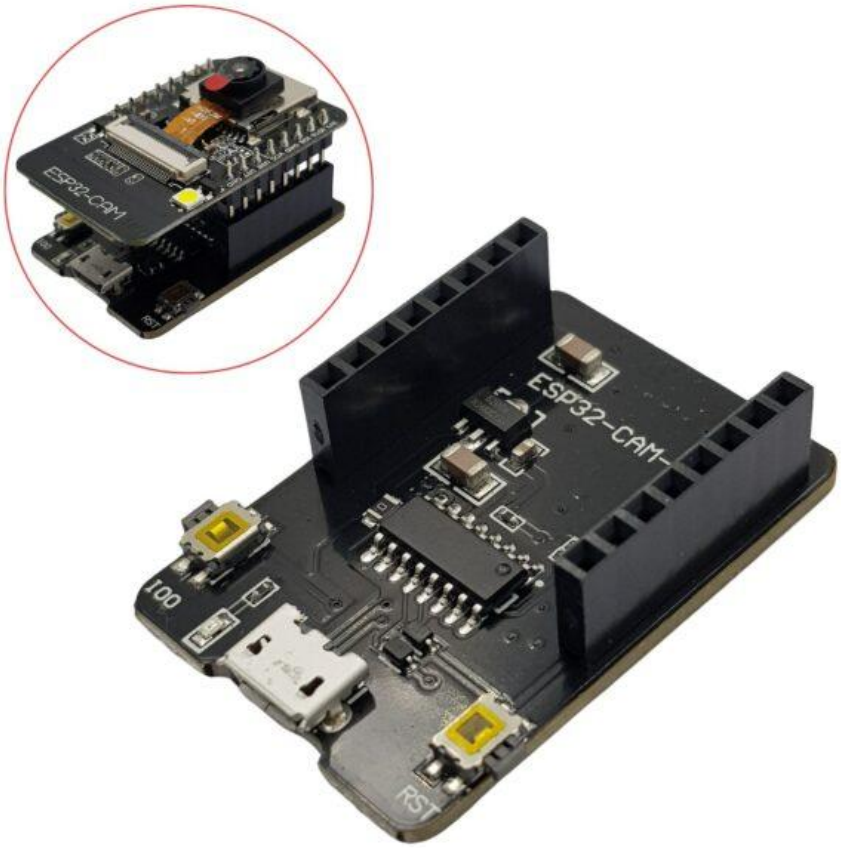
Hình 9 FTDI UART USB to TTL

- Thông số kĩ thuật:
  + IC chính: FT232RL chính hãng FTDI
  + Nguồn cấp: 5VDC từ cổng USB (cổng mini USB hoặc USB Type-C)
  + Có ngõ ra nguồn có thể điều chỉnh 3V3 hoặc 5VDC
  + Chuyển giao tiếp từ USB sang UART TTL
  + Drive hỗ trợ Windows Mac, Linux
  + Có cầu chì tự phục hồi: 500mA
  + Tốc độ Baudrate: tùy chỉnh
  + Kích thước PCB: 36 x 18.5mm
  + Trọng lượng: 3g

g. Đế nạp chương trình ESP32-CAM micro USB

- Đế nạp chương trình ESP32-CAM là công cụ hỗ trợ việc lập trình và nạp firmware cho module ESP32-CAM một cách nhanh chóng và dễ dàng. Thiết kế nhỏ gọn, tiện dụng, đế nạp này giúp loại bỏ các kết nối dây phức tạp, mang lại sự ổn định và hiệu quả cao khi phát triển các ứng dụng sử dụng ESP32-CAM.

+ Tích hợp cổng giao tiếp USB: Sử dụng cổng USB để kết nối trực tiếp với máy tính, tương thích với hầu hết các hệ điều hành như Windows, macOS, và Linux.
+ Chân kết nối tiêu chuẩn: Đế được thiết kế với các chân cắm tương thích hoàn hảo với module ESP32-CAM, đảm bảo quá trình nạp chương trình diễn ra ổn định.
+ Hỗ trợ các công cụ lập trình phổ biến: Dễ dàng sử dụng với Arduino IDE, PlatformIO hoặc các phần mềm lập trình tương tự.



Hình 10 Chân đế nạp code ESP 32 CAM

- Thông số kĩ thuật:
  + Model: ESP32-CAM
  + Tích hợp IC ổn áp nguồn 3.3VDC
  + Nguồn: 5VDC từ cổng Micro USB
  + IC chuyển giao tiếp USB-UART: CH340
  + Nút nhấn: RST và IO0
  + Kích thước: 27x40mm
  + Khối lượng: 7g
- Nguyên lý:

+ Chân đế được tích hợp 1 chip chuyển đổi (FTDI), chip này nhận dữ liệu từ cổng Type-C / USB, và chuyển đổi nó thành 2 đường tín hiệu UART:
+ TX (Transmit): truyền dữ liệu từ may tính sang ESP 32 CAM.
+ RX (Receive): Nhận dữ liệu từ ESP 32 CAM và gửi về máy tính (thường dùng để xem kết quả gỡ lỗi qua Serial Monitor).

- Quá trình nạp code:
  + Nhấn giữ nút GIPO 0.
  + Nhấn giữ nút RESET.
  + Nhả nút GIPO 0.
  + Lúc này, ESP 32 CAM đã vào Bootloader Mode và sẵn sàng nhận code từ máy tính qua giao tiếp UART.

III. Phân tích yêu cầu

1. Mục tiêu và phạm vi của hệ thống

a. Mục tiêu hệ thống

- Vấn đề thực tế: Khóa cửa cổ điển, mặc dù đã được sử dụng phổ biến trong thời gian dài, hiện nay đang bộc lộ nhiều hạn chế trong bối cảnh nhu cầu bảo mật và tự động hóa ngày càng cao. Việc phụ thuộc hoàn toàn vào chìa khóa vật lý khiến người dùng dễ gặp rắc rối khi làm mất hoặc quên chìa, thậm chí chìa có thể bị sao chép một cách dễ dàng, gây ra nguy cơ mất an toàn. Hơn nữa, khóa cơ học không có khả năng ghi lại lịch sử truy cập hay xác định ai đã mở cửa và vào thời điểm nào, điều này gây khó khăn cho việc quản lý trong các môi trường như văn phòng, chung cư hay nhà thông minh. Quá trình cấp hoặc thu hồi quyền truy cập cũng thủ công, mất thời gian và thiếu linh hoạt. Ngoài ra, khóa cổ điển không thể tích hợp với các hệ thống tự động hóa hoặc thiết bị IoT khác, khiến người dùng không thể giám sát, điều khiển từ xa hay kết hợp với các tính năng an ninh nâng cao. Dù có độ bền cơ học nhất định, khóa truyền thống vẫn có thể bị phá bằng các công cụ chuyên dụng và hoàn toàn thiếu các lớp bảo mật thông minh như sinh trắc học hay xác thực đa yếu tố. Chính vì những hạn chế này, nhu cầu chuyển đổi sang các hệ thống khóa thông minh nhận diện khuôn mặt đang trở nên tất yếu, nhằm mang lại sự tiện lợi, an toàn và quản lý hiệu quả hơn trong kỷ nguyên IoT hiện nay.
- Mục tiêu dự án Iot này mang lại: hệ thống khóa cửa thông minh nhận diện khuôn mặt ứng dụng IoT là tạo ra một giải pháp bảo mật hiện đại, tiện lợi và tự động hóa cao nhằm thay thế cho các loại khóa truyền thống. Cụ thể, dự án hướng đến việc tăng cường an toàn thông qua công nghệ nhận diện khuôn mặt – giúp đảm bảo chỉ những người được cấp quyền mới có thể mở khóa, loại bỏ nguy cơ bị sao chép chìa khóa hoặc mất thẻ từ. Bên cạnh đó, hệ

thống còn cho phép kết nối Internet, giúp người dùng giám sát và điều khiển cửa từ xa thông qua ứng dụng di động hoặc trình duyệt web, đồng thời nhận cảnh báo tức thời khi có truy cập trái phép. Ngoài ra, dự án cũng nhằm mục tiêu nâng cao khả năng quản lý và mở rộng, thông qua việc lưu trữ và xử lý dữ liệu trên nền tảng đám mây, cho phép người quản trị dễ dàng thêm, xóa hoặc cập nhật quyền truy cập của người dùng. Về mặt học thuật và kỹ thuật, dự án giúp người phát triển hiểu rõ hơn về công nghệ IoT, hệ thống nhúng, giao tiếp mạng, xử lý ảnh và trí tuệ nhân tạo, từ đó góp phần ứng dụng vào thực tế, phục vụ cho xu hướng nhà thông minh và tự động hóa trong thời đại 4.0.

b. Phạm vi triển khai

- Số lượng thiết bị (Mô hình nhà 1 cửa _ gia đình):
  + ESP32-CAM: 1.
  + Relay Module (1 kênh): 1
  + Khóa điện tử: 1.
  + Nguồn 5V + nguồn 12V: 1 bộ
  + FTDI USB-TTL: 1
- Môi trường hoạt động:
  + Điều kiện vật lý: Ngoài trời, cần vỏ kín chống bụi, chống nước, chống ngưng tụ và cách điện. Nếu lắp nơi có điều kiện khắc nhiệt hơn, cần phần cứng chịu nhiệt / vị trí che chắn.
  + Nguồn điện:
  + ESP32- CAM: nguồn 5V DC, >= 2A.
  + Khóa: Thường là 12V DC, cần cấp nguồn riêng hoặc bộ nguồn dùng chung có phân nhánh.
  + Bảo vệ nguồn: dùng fuse, diode flyback cho coil, transient suppressor (TVS) nếu đường dài.
  + Mạng và kết nối:
  + Wifi: RSSI tối ưu > -70 dBm; nếu yếu cần AP gần/mesh wifi.
  + Băng thông: stream camera MJPEG/ snapshot — định kỳ gửi ảnh lớn.
  + Độ tin cây: router/AP ổn định, QoS cho MQTT nếu nhiều thiết bị.

c. Tiêu chi thành công(KPIs)

- Độ chính xác: Sai số nhận diện khuôn mặt < 2%.
- Độ trễ: Ghi nhận khuôn mặt và gửi lên server trong vòng < 5s.
- Độ tin cậy: Tỉ lệ truyền dữ liệu thành công > 95%, hoạt động ổn định trong thời gian dài, chịu được các yếu tố của môi trường, thông báo khi xảy ra lỗi.

- Tiết kiệm tài nguyên: Tôi ưu hóa việc sử dụng năng lượng, bộ nhớ, băng thông khi truyền dữ liệu.
- Chi phí: Giảm thiểu chi phí lắp đặt, bảo trì và vận hành, hướng đến 1 giải pháp khả thi, dễ triển khai cho đến cả các hộ gia đình và doanh nghiệp nhỏ.
- Khả năng mở rộng: Thiết kế hệ thống có cấu trúc linh hoạt, dễ dàng thêm mới các tính năng trong tương lai.

d. Kết quả mong đợi

- An toàn trong việc bảo vệ tài sản của gia đình và doanh nghiệp.
- Có thểm giám sát từ xa qua thiết bị di động.
- Dữ liệu được truyền ngay lập tức, cảnh báo khi có hành vi không an toàn.

2. Mô tả tổng quan

a. Bối cảnh vấn đề

- Trong khuôn khổ gia đình và các doanh nghiệp có nhiều các vật dụng đáng giá mà bảo mật chứ tối ưu.
- Hiện nay, việc bảo mật an ninh thủ công -> để bị khai thác bởi những kẻ gian do không thể kiể soát được, thiệt hại về kinh tế.
- Đề xuất một thiết bị Iot có khả năng bảo về an toàn cho gia định và doanh nghiệp, vừa có khả năng giám sát từ xa.

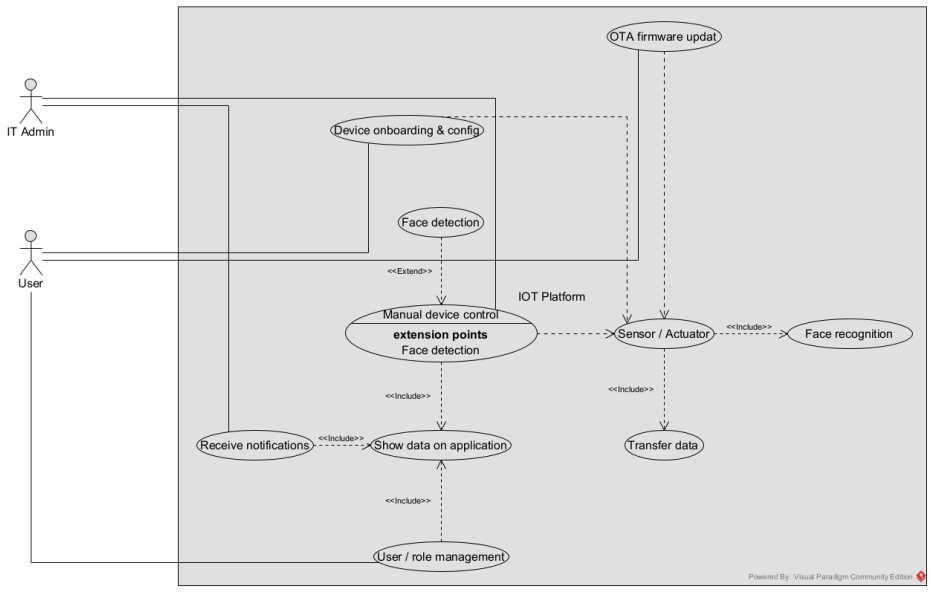
b. Yêu cầu từ các bên liên quan

- Người dùng cuối:
  + Hệ thống dễ sử dụng, thao tác mở/khóa cửa nhanh chóng và thân thiện.
  + Đảm bảo an toàn và bảo mật cao, tránh truy cập trái phép.
  + Có thể điều khiển và giám sát từ xa qua điện thoại hoặc máy tính.
  + Cung cấp thông báo thời gian thực (real-time) khi cửa mở, đóng hoặc phát hiện truy cập bất thường.
  + Hệ thống hoạt động ổn định, ít lỗi, độ trễ thấp.
  + Thiết bị có thiết kế gọn gàng, thẩm mỹ, phù hợp với không gian nhà ở.
- Doanh nghiệp, quản lý:
  + Dễ dàng triển khai, lắp đặt và bảo trì cho khách hàng.
  + Có thể quản lý và theo dõi thiết bị của nhiều khách hàng thông qua hệ thống quản trị tập trung (dashboard).
  + Hỗ trợ tích hợp mở rộng với các dịch vụ IoT khác (nhà thông minh, camera giám sát, cảm biến chuyển động…).
  + Chi phí sản xuất và vận hành hợp lý, có khả năng thương mại hóa.
  + Có hệ thống cập nhật phần mềm và bảo mật từ xa (OTA) để giảm chi phí bảo trì trực tiếp.

- Kỹ thuật, IT:
  + Thiết kế hệ thống có kiến trúc linh hoạt, dễ mở rộng và nâng cấp.
  + Hỗ trợ giao tiếp giữa các module (MCU, cảm biến, bộ truyền thông) một cách hiệu quả, tiết kiệm năng lượng.
  + Đảm bảo độ tin cậy, độ chính xác và độ trễ thấp khi xử lý tín hiệu điều khiển.
  + Áp dụng các giao thức truyền thông IoT phổ biến (như MQTT, HTTP hoặc WebSocket).
  + Tích hợp hệ thống bảo mật dữ liệu và xác thực người dùng.
  + Có công cụ giám sát, ghi log và xử lý sự cố để dễ dàng bảo trì sau này.

3. Yêu cầu chức năng

- Các chức năng cần có:
  + Thu thập dữ liệu từ cảm biến.
  + Gửi dữ liệu về gateway/cloud.
  + Lưu trữ và phân tích dữ liệu.
  + Hiển thị dữ liệu qua ứng dụng.
  + Điều khiển / ra lệnh ngược lại thiết bị.
  + Cập nhật Firmware
  + Quản lý người dùng và quyền truy cập
  + Nhận thông báo
  + Quản lý đăng ký khuôn mặt
  + Nhận diện khuôn mặt
- Đặc tả luồng công việc:
  + Use case:



Hình 11 Mô hình Use case tổng quan cho toàn bộ hệ thống

4. Yêu cầu phi chức năng

- Hiệu năng:
    + Độ trễ phản hồi: Thời gian từ khi người dùng gửi lệnh mở/khóa cửa đến khi cửa phản hồi không vượt quá 2 giây trong điều kiện mạng ổn định (<100ms ping).
    + Tốc độ xử lý tín hiệu tại thiết bị: Vi điều khiển phải xử lý tín hiệu điều khiển trong vòng <100ms kể từ khi nhận lệnh.
    + Dung lượng lưu trữ log: Thiết bị lưu trữ tối thiểu 100 bản ghi truy cập gần nhất (thời gian, người dùng, trạng thái cửa).
    + Tần suất gửi dữ liệu: Gói tin trạng thái được gửi lên server 10s 1 lần hoặc ngay khi có khi có thay đổi trạng thái.
- Bảo mật:
    + Truyền thông an toàn: Dữ liệu giữa thiết bị và máy chủ được mã hóa bằng TLS 1.2 hoặc cao hơn.
    + Xác thực người dùng: Yêu cầu mật khẩu >= 8 ký tự, chữ hoa, chứ thường, số, ký tự đặc biệt. Phiên đăng nhập tự động hết hạn sau 15’ hoạt động.
    + Quản lý truy cập: Chỉ ngươi dùng có quyên truy cập hợp lệ mới có thể điều khiển thiết bị. Ghi lại lịch sử truy cập.
- Độ tin cậy:
    + Thời gian hoạt động: Hệ thống đảm bảo 98% thời gian hoạt động mỗi tháng.
    + Khả năng phục hồi: Sau khi mất kết nối mạng, hay mất điện hệ thống tự khởi động và kết nối lại trong vòng <= 20s.
    + Xử lý lỗi: Hệ thống ghi log lỗi, tự động gửi cảnh báo lên server khi gặp sự cố. Dữ liệu không bị mất trong trường hợp gian đoạn mạng ngắn hạn (< 30s).
- Khả năng mở rộng:
    + Hệ thống máy chủ hỗ trợ tối thiểu 1000 thiết bị hoạt động đồng thời.
    + Có thể hoạt động trên 10000 thiết bị mà không cần thay đổi kiến trúc hệ thống, chỉ cần mở rộng tài nguyên phân cứng.
    + Có thể hỗ trợ thêm các module Iot khác thông qua API mở
- Chi phí & năng lượng:
    + Nguồn điện:
        - Nguồn điện cho toàn bộ thiết bị khóa cửa IoT cần đảm bảo ổn định, liên tục và an toàn.
        - Hệ thống thường sử dụng nguồn một chiều (DC) 5V hoặc 12V.
        - Nếu cấp điện liên tục từ lưới 220V, cần dùng biến áp và mạch ổn áp (adapter chuyển 220V AC → 5V/12V DC) để tránh quá tải hoặc dao động điện áp gây hỏng linh kiện.

+ Băng thông:
  - Triển khai thực tế, cần đăng kí dịch vụ lưu trữ của 1 bên thứ 3, tùy vào gói đăng kí sẽ nhận được lưu lượng tương ứng để lưu trữ hình ảnh thu nhận trong 1 khoảng thời gian xác định (tối thiểu 50MB / ngày).
+ Chi phí hạ tầng:
  Khoảng 20.000vnd / thiết bị / tháng, ở mức 5000 thiết bị -> Dùng MQTT + serverless analytics cho workload biến động.

5. Ràng buộc về kỹ thuật và môi trường

a. Môi trường hoạt động

- Nhiệt độ & độ ẩm: cảm biến đặt ngoài trời, chịu được biên độ -10°C đến 45°C, độ ẩm > 90%. → Cần chọn thiết bị chống nước IP67, vỏ bọc chống bụi.
- Nhiễu sóng:
  + Thiết bị IoT thường dùng Wi-Fi / Bluetooth / RF 2.4 GHz, dễ bị nhiễu bởi:
  + Lò vi sóng, router Wi-Fi khác, thiết bị Bluetooth, hoặc nguồn cao tần.
  + Cần bố trí khoảng cách tối thiểu 0.5–1 m với các thiết bị gây nhiễu mạnh.
  + Thiết kế mạch và vỏ kim loại cần có lớp chống nhiễu (EMC shielding) để đảm bảo tín hiệu ổn định.
  + Có thể áp dụng lọc nguồn (LC filter) và tách mass digital/analog để giảm nhiễu.
- Nguồn cấp:
  + Nguồn điện cần ổn định và bảo vệ quá áp:
  + Nguồn vào: 220 V AC (nguồn lưới).
  + Nguồn ra cấp cho thiết bị: 5 V DC (vi điều khiển, cảm biến), 12 V DC (motor/servo).
  + Sử dụng biến áp cách ly hoặc adapter switching (220 V → 12 V/2A) để tránh quá tải.
  + Cần tụ lọc (100 µF – 470 µF) và diode bảo vệ ngược cực trong mạch nguồn.
  + Trong trường hợp mất điện, nên có pin dự phòng hoặc UPS mini (5 V – 12 V) đảm bảo hệ thống hoạt động 2–6 giờ.

b. Ràng buộc pháp lý

- Tần số vô tuyến
  + Thiết bị sử dụng Wi-Fi (2.4GHz hoặc 5GHz), Bluetooth, hoặc các giao thức truyền thông tầm ngắn khác phải hoạt động trong dải tần được cấp phép bởi Bộ Thông tin & Truyền thông Việt Nam (MIC).

+ Không được phát công suất vượt quá giới hạn cho phép (≤ 100 mW đối với Wi-Fi 2.4GHz).
+ Khi triển khai thực tế, thiết bị cần được kiểm định tương thích điện từ (EMC) và có tem chứng nhận hợp quy (CR).

- Bảo mật dữ liệu
  + Dữ liệu khuôn mặt, tên người dùng, và lịch sử truy cập là thông tin nhạy cảm theo Nghị định 13/2023/NĐ-CP về bảo vệ dữ liệu cá nhân.
  + Việc thu thập, lưu trữ và xử lý dữ liệu phải có sự đồng ý rõ ràng của người dùng.
  + Dữ liệu phải được mã hóa trong quá trình lưu trữ và truyền tải (AES, SSL/TLS).
  + Người dùng có quyền yêu cầu chỉnh sửa, xóa dữ liệu hoặc rút lại sự đồng ý bất cứ lúc nào.
- An ninh mạng
  + Tuân thủ Luật An ninh mạng 2018: bảo đảm an toàn thông tin khi kết nối Internet, không để lộ lỗ hổng gây truy cập trái phép.
  + Hệ thống phải có cơ chế xác thực và phân quyền truy cập (admin, người dùng, khách).
  + Phải có biện pháp chống tấn công mạng, như tường lửa phần mềm, chống giả mạo yêu cầu (CSRF, XSS), và giám sát đăng nhập bất thường.

c. Tài nguyên thiết bị

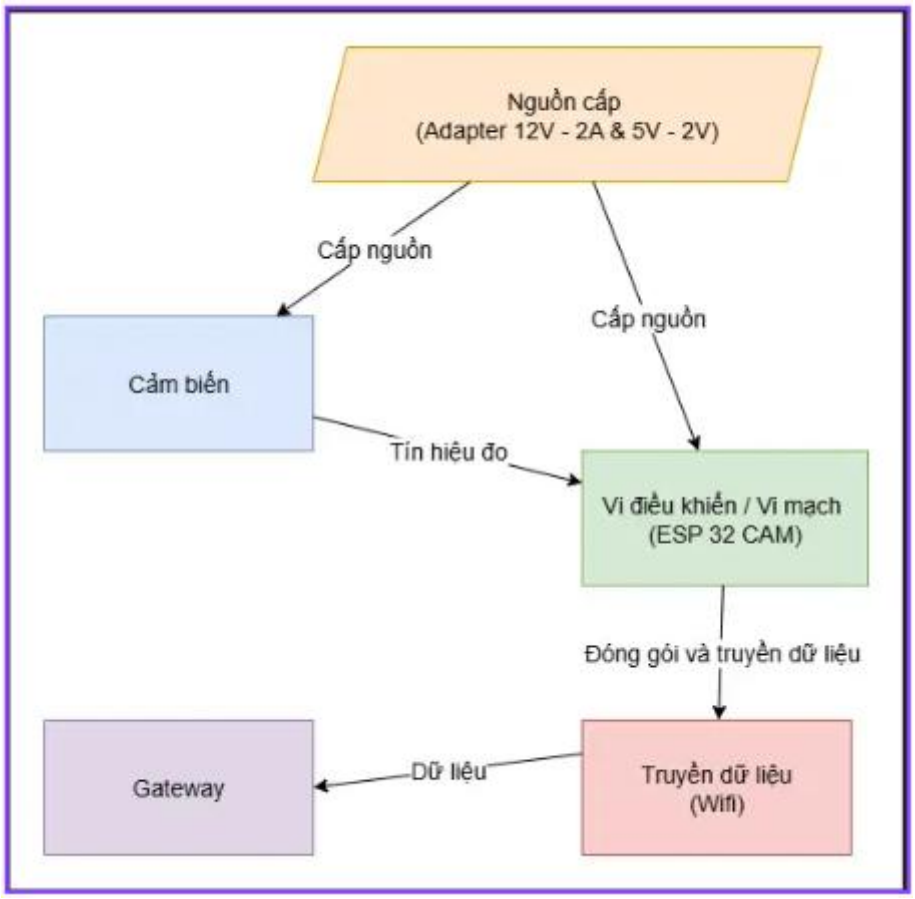
- Hệ thống sử dụng vi điều khiển (như ESP32) với CPU ≥ 160 MHz**,** RAM ≥ 300 KB**,** Flash ≥ 8 MB để đảm bảo xử lý nhận diện khuôn mặt ổn định.

  Nguồn cấp DC 5V–12V, dòng tiêu thụ khoảng 400–1000 mA, cần adapter ổn áp, chống nhiễu **và** pin dự phòng 2–6 giờ khi mất điện.

  Toàn hệ thống phải tối ưu tài nguyên để vừa đảm bảo hiệu năng, vừa tiết kiệm năng lượng và bảo vệ linh kiện.

IV. Phân tích thiết kế hệ thống

1. Thiết kế mức vật lý



Hình 12 Mô hình thiết kế hệ thống mức vật lý

- Phân tích các khối chức năng:
  + Nguồn cấp:
    - Nguồn cung cấp năng lượng cho các linh kiến trong thiết bị.
    - Nguồn 5V cấp cho ESP 32 CAM, Nguồn 12 V cấp cho khóa và relay.
    - Cảm biến:
    - Trong dự án sử dụng nút bấm thay cho các cảm biến chuyển động, nhiệt, …
    - Điều khiển esp 32 cam thực hiện ghi nhận hình ảnh gửi về server.
  + Vi điều khiển:
    - Sử dụng ESP 32 CAM.
    - Bộ não trung tâm điều khiển các linh kiến khác trong hệ thống.
    - Nhận nguồn 5V, thu ảnh gửi lên server và nhận tín hiệu xử lý khi kiểm tra khuôn mặt hoàn tất.
    - Đóng gói dữ liệu đưa vào giao thức MQTT để chuẩn bị gửi đi.
  + Truyền dữ liệu (Wifi):

- Đây thực chất là module Wifi tích hợp bên trong ESP 32 CAM được vẽ tách ra để nhấn mạnh chức năng truyền thông.
- Vai trò: Vận chuyển gói tin (ảnh + thông tin thiết bị) qua sóng vô tuyến.

+ Gateway:
- Mô tả: Cổng kết nối mạng.
- Trong thực tế: Đây chính là các modem Wifi / Router tại nhà.
- Vai trò: Nhận tín hiệu từ ESP 32 CAM và chuyển gói tin đó vào môi trường Intenet để gửi đến Server.

2. Thiết kế phần mềm

a. Ngôn ngữ lập trình

- C++: Lập trình cho ESP32, xử lý các tín hiệu từ cảm biến và truyền dữ liệu.
- Python: Triển khai mô hình AI nhận diện khuôn mặt bằng thư viện OpenCV hoặc TensorFlow.
- Backend API: Node.js (Express) hoặc Python (Flask/FastAPI).
- Frontend: ReactJS
- Cơ sở dữ liệu: MongoDB

b. Giao thức truyền thông

- Giao tiếp phần cứng: UART, SPI, I2C, PWM, ADC, và DAC cho giao tiếp dữ liệu với các thiết bị ngoại vi, trong đó SPI được sử dụng để giao tiếp với module camera.
- Wi-Fi: Theo chuẩn IEEE 802.11 b/g/n (2.4GHz), dùng cho kết nối không dây trong mạng cục bộ và Internet.
- MQTT: Giao thức IoT tối ưu, truyền dữ liệu thời gian thực, độ trễ thấp, băng thông nhỏ, phù hợp thiết bị hạn chế tài nguyên.
- HTTP/HTTPS: Giao thức phổ biến cho trao đổi dữ liệu client–server, đảm bảo bảo mật và tương thích cao.

c. Công cụ phát triển

- Arduino IDE
- Visual Studio Code
- Python IDE (VS Code)
- Node.js / ExpressJs/ ReactJs
- MongoDB
- GitHub
- Ngoài ra còn có các thư viện AI (OpenCV, TensorFlow/Keras), Postman

d. Phần mềm

- Ứng dụng người dùng: Website
- Chức năng:

+ Hệ thống nhận diện khuôn mặt từ thiết bị IoT hoặc ứng dụng, đối chiếu với cơ sở dữ liệu để xác định danh tính.
+ Đăng ký khuôn mặt mới.
+ Quản lý thông tin ngươi dùng.
+ Cho phép mở khóa từ xa, vô hiệu hóa và cấp quyền truy cập.
+ Phát hiện bất thường và cảnh báo thời gian thực.
+ Cập nhật firmware từ xa.

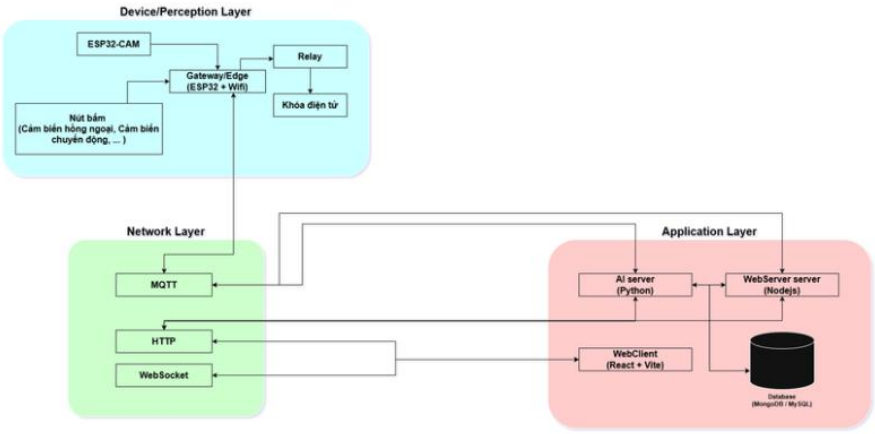
- Khả năng tích hợp: Restfull API, Microservices

3. Thiết kế Logic hệ thống

a. Yêu cầu và phạm vi logic

- Mục tiêu hệ thống:
  Hệ thống gồm 4 mục tiêu chính:
  + Giám sát: trạng thái khóa, lịch sử log ra vào.
  + Điều khiển: (Mục tiêu chính của hệ thống) => Mở / Khóa từ xa thông qua Server AI / Website quản trị.
  + Thu thập dữ liệu: Thu thập hình ảnh từ ESP 32 CAM phục vụ xác thực và lưu log truy cập.
  + Phân tích: Model phân tích hình ảnh từ ESP 32 để phục vụ xác thực người dùng.
- Thực thể cần quản lý:
  Hệ thống gồm 3 nhóm thực thể cần quản lý:
  + Thực thể người: Admin, User, Dữ liệu sinh trắc.
  + Thiết bị: ESP 32 CAM, Nút bấm, Khóa điện tử,…
  + Dữ liệu: Dữ liệu sinh trắc => vector đặc trưng khuôn mặt, log truy cập(trạng thái, thời gian, ảnh, …), Dữ liệu ngươi dùng, … (của website quản trị).
- Ràng buộc logic:
  + Tần suất dữ liệu: Hệ thống hoạt động dựa theo sự kiện, chỉ gửi ảnh khi nhân nút bấm.
  + Độ trễ:
    - Yêu cầu độ trễ khi giao tiếp với thiết bị (từ khi bấm nút -> xác thực thành công < 5s)
    - Độ trễ website quản trị: có thể chậm hơn.
  + An ninh: Xác thực thiết bị, Xác thực người dùng, mã hóa dữ liệu quan trọng như mật khẩu, dữ liệu sinh trắc.

b. Kiến trúc hệ thống



Hình 13 Kiến trúc 3 tầng cho hệ thống

- Mô tả:
  + Lớp thiết bị / Nhận thức (Device / Preception Layer):
    - Đây là nới phần cứng (hardware) hoạt động, tương ứng với phần thiết bị thực tế.
    - ESP 32 CAM: Đóng vai trò là camera, thu nhận dữ liệu hình ảnh.
    - Gateway / Edge (ESP 32 + Wifi): trong dự án, ESP 32 CAM đảm nhiệm vai trò này, kết nối Wifi để gửi dữ liệu đi.
    - Nút bấm / Cảm biến: Là các tác nhân kích hoạt. Khi bấm nút hoặc cảm biến phát hiện chuyển động, kích hoạt ESP 32 CAM.
    - Relay + khóa điện tử: Là bộ phận chấp hành, nhận lệnh từ ESP 32 CAM để thực hiện hàn động mở cửa.
  + Lớp mạng (Network Layer):
    - Đây là đường ống vận chuyển dữ liệu, Thiết kế sử dụng mô hình lai.
    - MQTT: Dùng giao tiếp giữa ESP 32 với Server.
    - Ưu điểm: Nhẹ, nhanh, giữ kết nối ổn định, rất phù hợp để gửi lệnh điều khiển tức thì.
    - HTTP: Dùng để WebClient giao tiếp với WebServer, hoặc để ESP 32 gửi file ảnh lớn (Nêu MQTT bị giới hạn dung lượng).
    - WebSocket: Dùng để WebServer cập nhật trạng thái thời gian thực xuống WebClient mà không cần reload lại trang web.

      Đây là đường ống vận chuyển dữ liệu,
  + Lớp Ứng dụng (Application Layer):

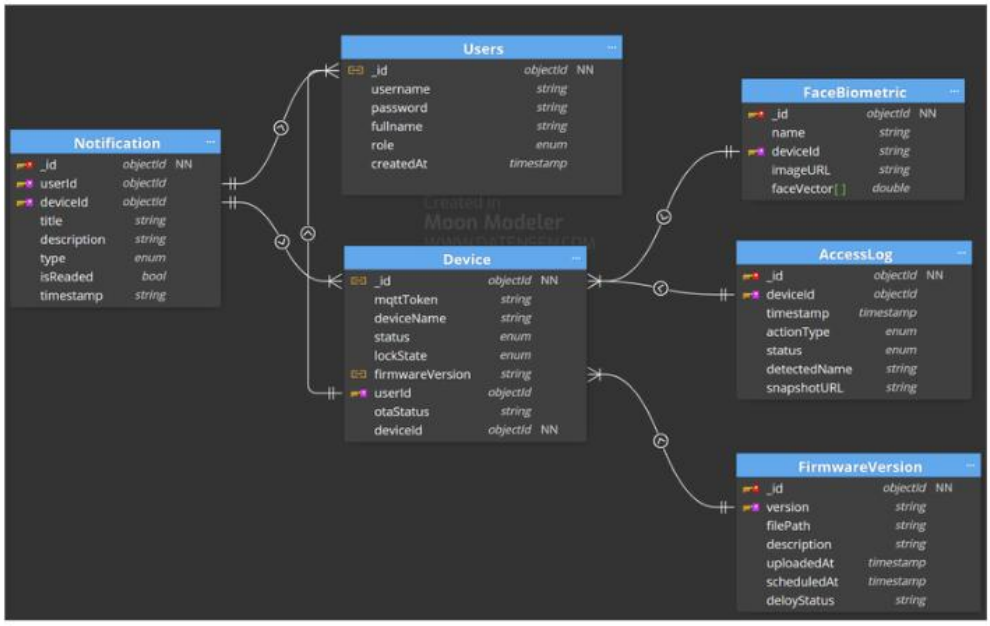
- Đây là lớp xử lý trung tâm, được chia theo kiến trúc Microservices:
- AI Server (Python): Dùng để xử lý dữ liệu hình ảnh.
- Python có các thư viện hỗ trợ mạnh (OpenCV, Face_recognation, Pytorch,…) tách riêng nó ra để nhận diện khuôn mặt.
- WebServer (Node js): dùng để quản lý logic nghiệp vụ:
- Nhận kết quả từ AI Server, kiểm tra quyền truy cập trong Database, quản lý người dùng, gửi lệnh xuống MQTT, Node js xử lý I / O rất nhanh.
- WebClient (React + Vite): Giao diện người dùng, để xem lịch sử ra vào, xem camera, hoặc mở cửa từ xa.
- Database (MongoDB): Lưu trữ dữ liệu thông tin người dùng, lịch sử log ra vào, đường dẫn ảnh.

- Luồng hoạt động:
  + Thu nhập: Người dùng nhấn nút ->ESP 32 CAM tỉnh dậy -> Chụp ảnh.
  + Truyền tải: ESP 32 CAM gửi ảnh qua giao thức mạng MQTT (hoặc HTTP cho các ảnh có dung lượng lớn) lên server.
  + Xử lý:
    - Dữ liệu đi vào AI Server để phân tích, trích xuất các đặc trưng.
    - Kết quả gửi về cho WebServer, server so sánh các đặc trưng này với database để đưa ra quyết định mở cửa hay không.
  + Ra quyết định:
    - Server nodejs kiểm tra Database. Nếu người dùng hợp lệ -> tạo lệnh mở cửa.
    - Server nodejs cũng đẩy thông báo qua WebSocket cho WebClient để thông báo cho chủ của khóa.
  + Điều khiển:
    - Server nodejs gửi lệnh “OPEN” qua giao thức MQTT cho ESP 32.
    - ESP 32 lắng nghe qua giao thức MQTT, nhận lênhj -> kích hoạt relay -> Mở khóa.
- Đánh giá:
  + Ưu điểm:
    - Tách biệt rõ ràng: Việc tách server AI và server Web riêng giúp hệ thống không bị treo. Nếu Ai đang trong quá trình xử lý ảnh, thì Webserver vẫn hoạt động để phục vị các người dùng bình thường.

- Tính năng mở rộng: Tương lai có thể mở rộng thêm nhiệu thiết bị nữa, hay thêm cảm biến thay thế cho nút bấm vật lý, chỉ cần cắm thêm thiết bị và đẩy vào MQTT Broker, không cần sửa lại cấu trúc của server.
- Trải nghiệm ngươi dùng tốt: Sử dụng React + Webserver giúp giao diện web mượt mà, hiệu năng cao, cập nhật tức thì.

+ Nhược điểm & Thách thức:
  - Độ phức tạp: Có cấu trúc theo mô hình Microservices, việc triển khai code phức tạp.
  - Độ trễ:
  - Ảnh từ ESP 32 CAM -> Server -> AI xử lý -> Webserver -> MQTT -> Relay.
  - Mỗi bước tốn 1 chút thời gian, với nhận diện khuôn mặt, tổng thời gian phải dưới 5s mới đạt hiệu quả trải nghiệm ngươi dùng.
  - Gửi ảnh qua MQTT: giao thức này không tối ưu khi gửi file lơn.

c. Database



Hình 14 Cơ sở dữ liệu triển khai hệ thống

- Mô tả:

+ Người dùng (Users):
  - Vai trò: Quản lý tài khoản đăng nhập vào ứng dụng (Website).
  - Các trường dữ liệu:

    - role: Phân quyền (admin / user). Admin có quyền quản lý thiết bị, còn User có quyền giám sát mở cửa.
    - _id: khóa chính liên kết với các bảng Device & Notification.

+ Thiết bị (Device):
  - Vai trò: Đại diện cho hệ thống mức vật lý (ESP 32 CAM + Relay + Khóa).
  - Các trường dữ liệu:
    - mqttToken: Đây là token riêng biệt để ESP 32 dùng khi kết nối vào MQTT Broker. Giúp bảo mật, tránh người lạ có thể kết nối đến topic điều khiên.
    - status: trạng thái của thiết bị (on / off) _ nhận biết thiết bị có được cấp nguồn hay không.
    - lockState: Trạng thái của khóa cửa (đang mở hay đóng).
    - otaStatus & firmwareVersion: Hỗ trợ tính năng cập nhật firmware từ xa (OTA) mà không cần cắm trực tiếp máy tính vào thiết bị.
    - userId: Xác định chủ sở hữu của thiết bị.

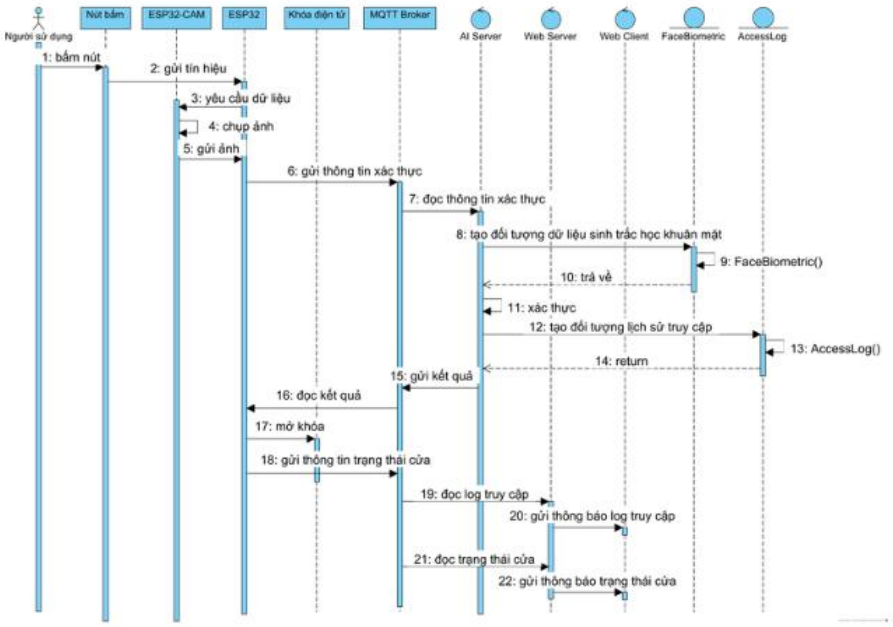
+ Dữ liệu khuôn mặt (FaceBiometric):
  - Vai trò: Lưu dữ liệu các vector đặc trưng của khuôn mặt đã được đăng ký với từng thiết bị phục vụ quá trình nhận diện khuôn mặt.
  - Các trường dữ liệu:
    - deviceId: xác định thiết bị mà khuôn mặt được cấp quyền mở khóa.
    - imageURL: Đường dẫn tới file lưu ảnh gốc của khuôn mặt đã đăng ký.
    - faceVector[]: mảng số thực, lưu các đặc trưng của khuôn mặt sau khi qua xử lý của mạng neural.

+ Nhật ký truy cập (AccessLog):
  - Vai trò: Lưu lại lịch sử truy cập phục vụ tra cứu và bảo mật.
  - Các trường dữ liệu:
    - actionType: Loại hoạt động (Mở của bằng khuôn mặt, bằng hệ thống, cảnh báo,…).
    - status: Thành công / thất bại.
    - detectedName: Nếu model nhận ra ai, nó sẽ lưu tên người đó vào trường này.
    - snapshotURL: đường dẫn tới ảnh chụp khoảng khắc đó. Là bức ảnh ESP 32 CAM chụp và gửi lên server nhận diện.

+ Thông báo (Notification):
  - Vai trò: lưu trữ các thông báo gửi đến client người dùng phổ thông.
  - Liên kết: Nối với Users và Device.
+ Phiên bản cập nhật (FirmwareVersion):
  - Vai trò: Quản lý các file mã nguồn để cập nhật cho ESP 32 CAM.
  - filePath: Đường dẫn tới file mã nguồn cập nhật.
  - deployStatus: trạng thái cập nhật.

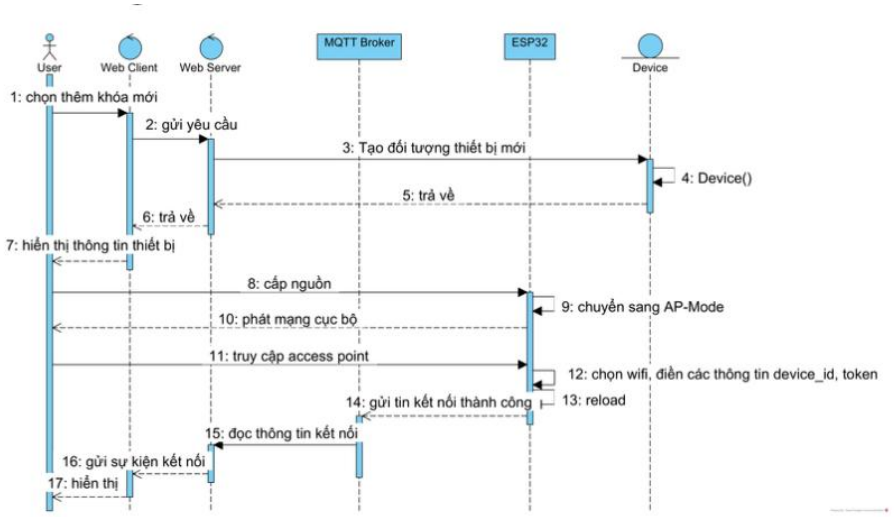
d. Sơ đồ luồng

- Xác thực khuôn mặt:
  1. Người dùng bấm nút trên thiết bị.
  2. Nút bấm kích hoạt chế độ cấp nguồn cho ESP 32.
  3. ESP 32 CAM kích hoạt camera.
  4. Camera chụp ảnh của người dùng.
  5. Camera gửi lại ảnh cho ESP 32.
  6. ESP 32 đóng gói và gửi thông tin xác thực đến MQTT Broker.
  7. MQTT Broker gửi dữ liệu thông tin xác thực đến AI server.
  8. AI Server gọi đến đối tượng FaceBiometric.
  9. FaceBiometric thực hiện trích xuất các đặc trưng của khuôn mặt và đóng gói đối tượng FaceBiometric.
  10. FaceBiometric gửi lại đối tượng về cho AI server.
  11. AI Server thực hiện xác thực thông qua đánh giá so sánh với các vector đặc trưng của các khuôn mặt đã được đăng kí với thiết bị.
  12. AI server gọi đến đối tượng AccessLog.
  13. AccessLog thực hiện tạo đối tượng AccessLog.
  14. AccessLog trả đối tượng AccessLog cho AI server.
  15. AI server gửi kết quả nhận diện về cho MQTT Broker.
  16. MQTT phát tín hiệu điều khiển ESP 32.
  17. ESP 32 nhận tín hiệu và thực hiện mở khóa.
  18. ESP đồng thời gửi trạng thái thiết bị lại cho MQTT Broker.
  19. MQTT Broker gửi log truy cập đến cho Web server.
  20. Web Server gửi thông báo log truy cập đến cho Web Client.
  21. MQTT Broker gửi trạng thái cửa đến cho Web server.
  22. Web server gửi thông báo trạng thái cửa cho Web Client.



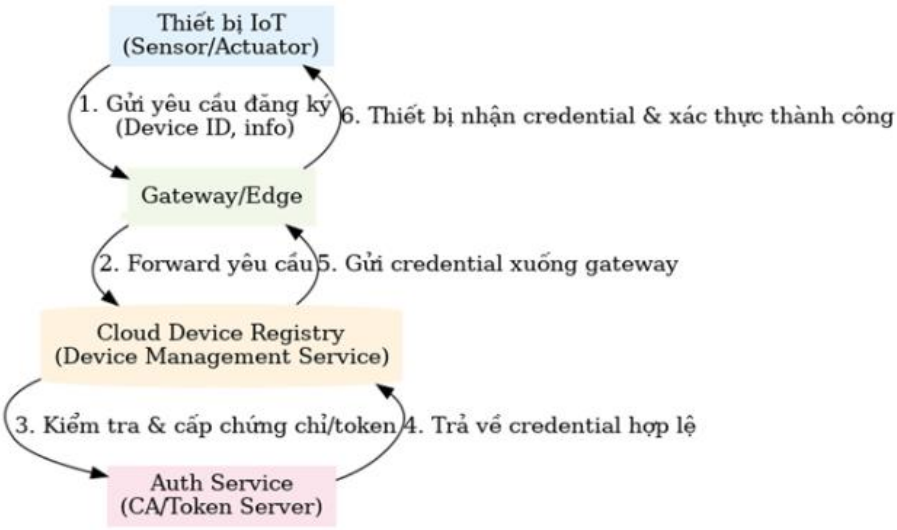
Hình 15 Sơ đồ luồng hoạt động module xác thực khuôn mặt

- Đăng ký thiết bị:
  1. Người dùng chọn thêm mới khóa trên giao diện Web Client.
  2. Web Client gửi yêu cầu đến cho Web Server.
  3. Web Server gọi đến Device.
  4. Device thực hiện việc tạo đối tượng Device.
  5. Device trả kết quả cho Web Server.
  6. Web Server trả kết quả cho Web Client.
  7. Web Client hiển thị thông tin của thiết bị mới cho người dùng.
  8. Người dùng cấp nguồn cho ESP 32.
  9. ESP 32 chuyển sang trạng thái AP Mode.
  10. ESP 32 phát mạng cục bộ.
  11. Người dùng truy cập vào access point.
  12. Người dùng chọn wifi, điền thông tin của device, token.
  13. ESP 32 cam thực hiện reset lại thiết bị để nhận thông tin mới nhất của người dùng.
  14. ESP 32 gửi thông báo kết nối thành cồng cho MQTT Broker.
  15. MQTT đọc gói tin kết nối thanh công và chuyển cho Web Server.
  16. Web Server gửi sự kiện kết nối đến Web Client.
  17. Web Client hiện thị thông tin của thiết bị mới cho người dùng.



Hình 16 Sơ đồ luồng module đăng ký thiết bị

e. Thiết kế logic và an ninh



Hình 17 Kiến trúc logic và an ninh cho hệ thống

- Quy trình vận hành của thiết bị:
  + Gửi yêu cầu đăng kí:
    - Thiết bị IoT: Khi khởi động lần đầu (hoặc sau khi reset), thiết bị chưa có cài đặt khóa để mở cửa. Nó gửi 1 gói tin chứa thông tin định danh chứa thông tin phần cứng lến Gateway.
  + Chuyển tiếp yêu cầu:
    - Gateway / Edge: Đóng vai trò trung chuyển. Nó nhận yêu cầu từ thiết bị và chuyển tiếp đến Server quản lý.
  + Kiểm tra & cấp chứng chỉ / token:
    - Server kiểm tra xem thiết bị này có nằm trong database không.

- Nếu hợp lệ, nó yêu cầu Auth Service tạo ra 1 khóa riêng cho thiết bị này.

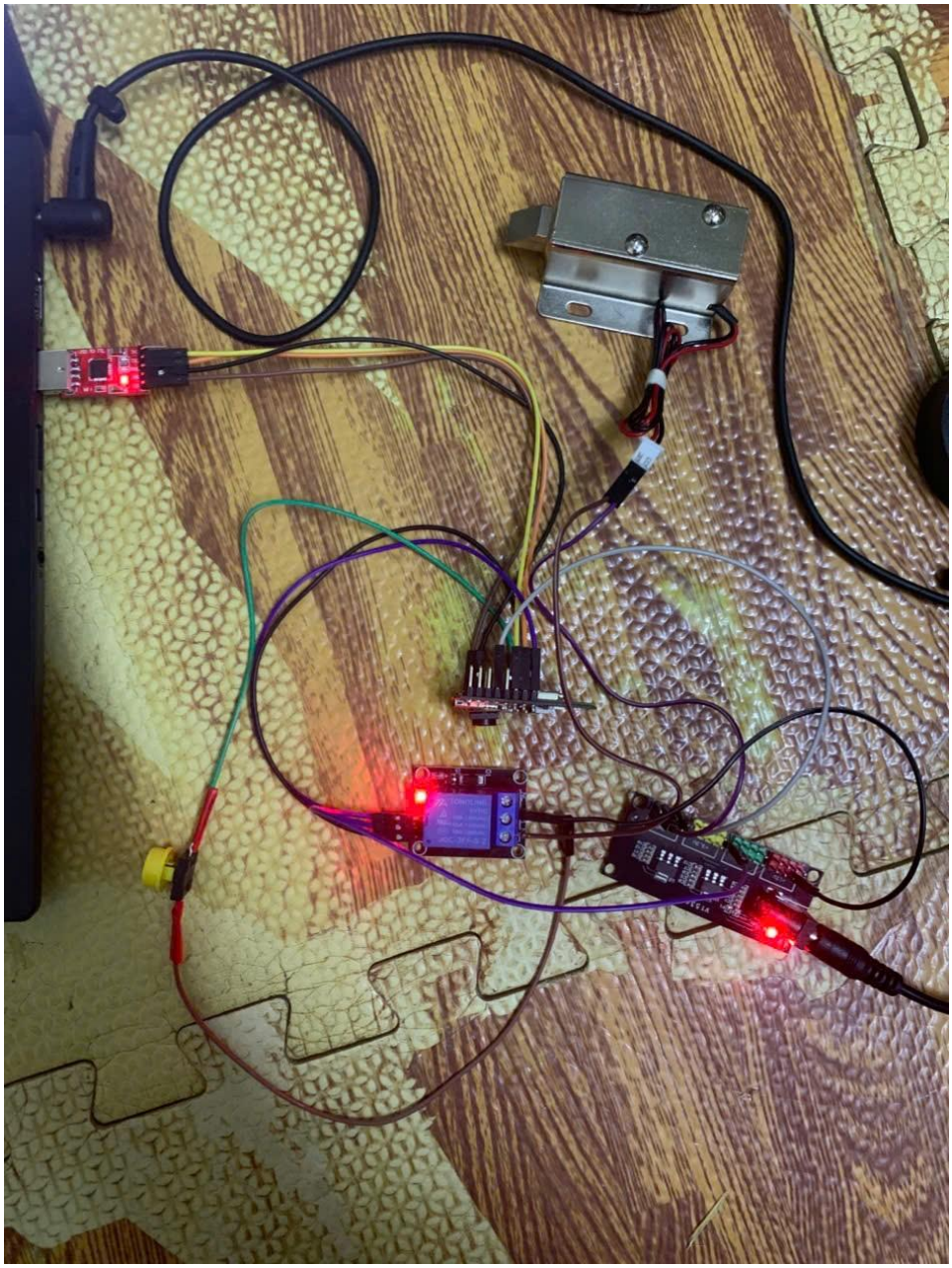
+ Trả Credential hợp lệ:
  - Auth Service: Tạo ra 1 token hoặc certificate và trả lại cho registry.
+ Gửi Credential lại cho Gateway:
  - Server gửi lại token cho gateway.
+ Thiết bị nhận Crediental & xác thực thành công
  - Gateway truyền token cho ESP 32.
  - ESP 32 lưu token này vào bộ nhớ (Flash).

V. Đánh giá kết quả dự án

1. Kết quả triển khai

a. Phần cứng

- Nguồn:
  - +12V: Relay VCC, COM
  - +5V: ESP 32 CAM 5V
  - GND: Khóa, ESP 32 CAM GND, Button, Relay GND
- Relay:
  - IN: ESP 32 CAM IO14
  - GND: GND nguồn
  - VCC: +12V nguồn
  - NO: Khóa
  - COM: +12V nguồn
- ESP 32 CAM:
  - +5V: +5V nguồn
  - GND: GND nguồn
  - IO14: IN Relay
  - IO13: Button



Hình 18 Kết nối mạch vật lý

b. Phần mềm

2. Đánh giá kết quả

a. Về mặt chức năng

- Chức năng nhận diện: Camera thu thập hình ảnh tốt, module nhận diện phân biệt được khuôn mặt đã đăng ký (Admin/User) và khuôn mặt người lạ (Unknown).
- Chức năng điều khiển khóa: Khi nhận diện đúng khuôn mặt, tín hiệu điều khiển relay được kích hoạt ngay lập tức, chốt cửa mở ra. Khi nhận diện sai, chốt cửa vẫn giữ nguyên trạng thái đóng.
- Giao diện Web: Server hoạt động ổn định, cho phép xem video stream (luồng hình ảnh) thời gian thực với độ trễ thấp. Các nút

chức năng "Mở khóa", "Thêm khuôn mặt" trên giao diện hoạt động chính xác.

- Nút bấm vật lý: Nút nhấn mở cửa từ bên trong hoạt động nhạy, ngắt chương trình nhận diện tạm thời để ưu tiên mở cửa thủ công.

b. Về mặt hiệu năng

- Khoảng cách nhận diện tối đa: Tốt nhất từ 20 – 40 cm, khoảng cách phù hợp cho người dùng đứng trước cửa. Khoảng cách xa hơn / gần hơn camera không thể thu nhận toàn bộ khuôn mặt.
- Tốc độ xử lý: Thời gian từ lúc đưa mặt vào camera đến lúc nhận diện xong: Trung bình khoảng 3 – 5 s (phụ thuộc vào chất lượng mạng Wi-Fi và độ phức tạp của ảnh).
- Điều kiện ánh sáng: Ánh sáng ban ngày/trong phòng: Hệ thống hoạt động tốt, độ chính xác đạt khoảng 85-90%. Ánh sáng yếu/Ban đêm: Khả năng nhận diện giảm đáng kể do nhiễu ảnh (Noise). Đây là hạn chế chung của cảm biến quang học thông thường khi không có đèn hồng ngoại hỗ trợ. Ngược sáng: Khi có ánh sáng mạnh chiếu thẳng vào camera từ phía sau người dùng, khuôn mặt bị tối và khó nhận diện.

c. Kết quả kiểm thử

STT	Kịch bản kiểm thử (Test Case)	Dữ liệu đầu vào	Kết quả mong đợi	Kết quả thực tế	Đánh giá
1	Người dùng đã đăng ký	Khuôn mặt ID: SinhVien1	Mở khóa + Thông báo "Welcome"	Khóa mở, LED xanh sáng	Đạt
2	Người lạ chưa đăng ký	Khuôn mặt người lạ	Không mở khóa + Cảnh báo	Khóa đóng, LED đỏ sáng	Đạt
3	Không có người	Không có khuôn mặt	Chế độ chờ (Idle)	Hệ thống chờ	Đạt
4	Che một phần mặt	Đeo khẩu trang/kính râm	Không nhận diện hoặc yêu cầu thử lại	Không nhận diện được	Chấp nhận được
5	Nhấn nút mở cửa	Nhấn nút Button	Mở khóa ngay lập tức	Khóa mở	Đạt

VI. Kết luận

1. **Kết quả đạt được**
   a. **Về mặt lý thuyết**
      Qua quá trình nghiên cứu và thực hiện đề tài, nhóm đã nắm vững kiến thức về vi điều khiển (như ESP32/ESP32-CAM), cách thức giao tiếp với module camera (OV2640) và nguyên lý hoạt động của các thuật toán xử lý ảnh, nhận diện khuôn mặt. Nhóm cũng đã hiểu rõ cơ chế truyền tải dữ liệu hình ảnh qua giao thức HTTP/WebSocket và cách xây dựng Web Server để quản lý hệ thống, đảm bảo sự tương tác mượt mà giữa phần cứng và giao diện người dùng.
   b. **Về mặt thực tiễn**
      Hệ thống đã hoạt động ổn định với các chức năng cơ bản: tự động nhận diện khuôn mặt đã đăng ký để mở khóa chốt điện từ và gửi hình ảnh người lạ về máy chủ để giám sát. Tốc độ nhận diện đạt mức chấp nhận được, giúp người dùng không cần mang theo chìa khóa cơ, giảm thiểu rủi ro thất lạc chìa khóa. Giao diện quản lý trực quan giúp người dùng dễ dàng theo dõi lịch sử ra vào và kiểm soát trạng thái cửa từ xa.
2. **Hạn chế**
   - Phụ thuộc vào điều kiện ánh sánh: Camera hoạt động chưa tốt trong môi trường thiếu sáng hoặc ngược sáng mạnh, làm giảm độ chính xác của thuật toán nhận diện khuôn mặt.
   - Độ trễ xử lý: Do giới hạn về tài nguyên phần cứng của vi điều khiển và tốc độ mạng, quá trình từ lúc chụp ảnh đến lúc mở cửa đôi khi còn có độ trễ nhất định, chưa đạt độ mượt mà tuyệt đối như các thiết bị thương mại cao cấp.
   - Phụ thuộc vào kết nối mạng: Khả năng gửi hình ảnh và giám sát từ xa phụ thuộc hoàn toàn vào sóng Wi-Fi. Nếu mất kết nối mạng, hệ thống chỉ có thể hoạt động ở chế độ offline (nếu có xử lý cục bộ) hoặc mất khả năng báo cáo sự cố.
     Khả năng chống mạo danh: Hệ thống hiện tại chưa có các biện pháp chuyên sâu để phân biệt giữa khuôn mặt thật và hình ảnh/video chất lượng cao (liveness detection), dẫn đến nguy cơ bị qua mặt bởi các thủ thuật giả mạo tinh vi.
3. **Hướng phát triển tương lai**
   a. **Về hệ thống**
      - Tích hợp cảm biến thay thế nút bấm vật lý: Bổ sung đèn LED hồng ngoại hoặc cảm biến chuyển động (PIR) để hệ thống hoạt động hiệu quả vào ban đêm và chỉ kích hoạt camera khi có người, giúp tiết kiệm năng lượng.

- Nguồn điện dự phòng: Nghiên cứu tích hợp pin sạc dự phòng hoặc hệ thống UPS nhỏ để đảm bảo khóa cửa vẫn hoạt động và duy trì an ninh khi xảy ra sự cố mất điện lưới.
- Đa dạng phương thức mở khóa: Phát triển thêm các phương thức dự phòng như mở bằng vân tay, thẻ từ RFID hoặc mật mã số trên ứng dụng di động để tăng tính linh hoạt khi nhận diện khuôn mặt gặp sự cố.
- Mở rộng kết nối: Hỗ trợ module SIM 4G/LTE để hệ thống gửi cảnh báo và hình ảnh ngay cả khi không có Wi-Fi.

b. Về bảo mật

- Mã hóa dữ liệu: Áp dụng các phương thức mã hóa đầu cuối (End-to-End Encryption) cho dữ liệu hình ảnh và tín hiệu điều khiển khi truyền tải qua mạng internet, ngăn chặn việc bị tin tặc đánh cắp luồng video hoặc can thiệp mở cửa trái phép.
- Bảo mật mức vật lý: Thiết kế vỏ hộp bảo vệ chắc chắn hơn, tích hợp cảm biến chống cạy phá (tamper switch) để phát báo động ngay khi có tác động vật lý nhằm phá hoại thiết bị.

VII. Tài liệu tham khảo

Ngô Diên Tập, Kỹ thuật vi xử lý và vi điều khiển, Nhà xuất bản Khoa học và Kỹ thuật, Hà Nội, 2019. (Dùng để tham khảo về cấu trúc vi điều khiển nói chung).

Phạm Minh Tuấn, Xử lý ảnh kỹ thuật số, Nhà xuất bản Khoa học và Kỹ thuật, 2018. (Dùng để trích dẫn các khái niệm về xử lý ảnh, điểm ảnh, lọc nhiễu...).

Bộ Thông tin và Truyền thông, Báo cáo chuyển đổi số và định hướng phát triển Internet vạn vật (IoT) tại Việt Nam, 2023. (Dùng cho phần mở đầu, nói về sự cấp thiết của đề tài).

Espressif Systems, "ESP32 Technical Reference Manual", Version 4.1, 2020. (Tài liệu gốc về chip ESP32 - bắt buộc phải có nếu dùng ESP32).

Viola, P. and Jones, M., "Rapid Object Detection using a Boosted Cascade of Simple Features", Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2001. (Đây là bài báo kinh điển về thuật toán phát hiện khuôn mặt Haar Cascade - nên đưa vào nếu bạn dùng thư viện nhận diện cơ bản).

F. Schroff, D. Kalenichenko and J. Philbin, "FaceNet: A Unified Embedding for Face Recognition and Clustering", IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015. (Tài liệu về Deep Learning trong nhận diện khuôn mặt - đưa vào nếu bạn muốn phần "Cơ sở lý thuyết" trông chuyên sâu hơn).

OmniVision, "OV2640 Color CMOS UXGA (2.0 MegaPixel) CameraChip Sensor Datasheet", Version 2.2. (Tài liệu về camera bạn sử dụng).